University of California
Santa Barbara

Three Applications of Gaussian Process Modeling in Evaluation of Longevity Risk Management

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Statistics & Applied Probability

by

James Kenneth Risk

Committee in charge:

Professor Michael Ludkovski, Chair, Chair
Professor Jean-Pierre Fouque
Professor Tomoyuki Ichiba

September 2017

The Dissertation of James Kenneth Risk is approved.

Professor Jean-Pierre Fouque

Professor Tomoyuki Ichiba

Professor Michael Ludkovski, Chair, Committee Chair

July 2017

Three Applications of Gaussian Process Modeling in Evaluation of Longevity Risk Management

by

James Kenneth Risk

I dedicate this dissertation to my brother, Justin Risk.

Acknowledgements

First, I would like to thank my advisor Prof. Mike Ludkovski for his help from the beginning of my PhD journey. The process of learning how to write academic papers was easy and efficient under his supervision. He always gave me invaluable feedback during all of our meetings, and served as an excellent role model of what I would one day like to become. I really appreciate the time he spent working with me.

I would also like to thank Prof. Jean-Pierre Fouque, of whom I had the most coursework with. He has an amazing way of presenting information and can turn the most complicated subject into an easy one with a few words and explanations. There is no way I would be as successful at probability theory and financial mathematics without his help. I also appreciate his kind demeanor – it is always easy to interact with him at conferences and other meetings.

I also express gratitude to Prof. Tomoyuki Ichiba who accepted to be a member of my dissertation committee. I also thank Prof. Chuck Akemann, who always had a positive attitude when I visited his office several hours a week when taking real analysis, Prof. Raya Feldman for always going out of her way to help others and for walking with me during my graduation ceremony, and Prof. Ian Duncan for our friendship as I TA'd his actuarial courses. In addition I want to express appreciation for my colleagues and friends at UCSB for their support during my PhD career. This especially includes to Brian Wainwright who was kind enough to let me stay at his guest house in my last year at UCSB since I lived off campus, and to Mostafa Mousavi who was an invaluable study partner going through all of our courses and studying for qualifying exams. I also thank Mark Dela who is

a good friend and gave me a warm welcome during our campus visit – it greatly influenced my decision to come to UCSB. Lastly, I thank Ruimeng Hu, Liangchen Li, Patricia Ning, and Yuri Saporito for their friendship during my studies.

Lastly, I want to thank my close family who was always there to support me and watched me grow into what I am today. This includes my parents, brother, grandma, and uncles and aunts who have always expressed encouragement and pride in me.

Curriculum Vitæ

James Kenneth Risk

Research Interest

Gaussian Processes

- Stochastic kriging, applications to quantile and level-set estimation, and probability of failure.

Computational Statistics

- Monte Carlo methods. Improving efficiency of existing methods via stochastic kriging.

Mathematical Finance & Actuarial Science

- Applications of Gaussian Processes regression to numerical BSDE methods and optimal stopping problems, and to pricing problems in insurance (including VaR estimation). Mortality modeling via Gaussian Processes.

Publications

(Preprints available at www.pstat.ucsb.edu/jrisk/research.html)

Risk, Jimmy, and Ludkovski, Michael. "Statistical emulators for pricing and hedging longevity risk products." *Insurance: Mathematics and Economics* 68 (2016): 45-60.

Risk, Jimmy, Ludkovski, Michael, and Zail, Howard. "Gaussian Process Models for Mortality Rates and Improvement Factors." *Submitted for Publication.* (May 2017) https://arxiv.org/abs/1608.08291

Risk, Jimmy, and Ludkovski, Michael. "Sequential Design Algorithms for Estimating Value-At-Risk for Longevity Risk." *To be submitted.*

Risk, Jimmy. "Correlations between Google search data and Mortality Rates." arXiv preprint arXiv:1209.2433 (2012). https://arxiv.org/abs/1209.2433

Education

Doctor of Philosophy, Statistics & Applied Probability
September 2013–July 2017
Emphasis in Financial Mathematics and Statistics

University of California, Santa Barbara, CA
Dissertation Committee:

- Michael Ludkovski (Advisor)
- Jean-Pierre Fouque
- Tomoyuki Ichiba

Thesis Topic: Applications of Gaussian Processes to Actuarial Modeling and Pricing
GPA: 3.90

Extended Adademic Visit
September 2015
ISFA: Institut de Science Financière et d'Assurances - Université Lyon 1
Topic: Stochastic Kriging in Longevity Risk Pricing
Invited by Stéphane Loisel

Master of Science, Statistics & Probability
January 2011–May 2013
Michigan State University, East Lansing, MI
GPA: 3.92

Bachelor of Science, Mathematics
January 2007–August 2010
Michigan State University, East Lansing, MI
Actuarial Specialization

Actuarial

Passed exams P, FM, MLC, C, MFE; All VEE credits completed

Graduate Coursework

Financial Mathematics

- Random Graph Theory & Financial Networks
- Backward Stochastic Differential Equations & Applications to Optimal Stopping and Control problems
- Numerical Methods in Mathematical Finance (Monte Carlo, Solutions of PDEs)
- Malliavin Calculus

- Stochastic Volatility Models
- Lévy Processes

Statistics & Probability

- Functional Regression
- Bayesian Analysis
- Probability Theory & Stochastic Calculus
- Statistical Theory

Mathematics

- Real Analysis, Introductory Functional Analysis, Spectral Theory

Teaching Experience

Teaching Associate
Department of Statistics & Applied Probability, University of California, Santa Barbara

- Lecturer for PSTAT 109 (Statistics for Economics) (Syllabus Link) (Summer 2015, 2016)
  Consists of directing entire course by self, including design of course syllabus and notes, lecturing, and delegating TA duties.

Department of Statistics & Probability, Michigan State University

- Lecturer for STT 200 (Introduction to Probability & Statistics) (Summer 2012)

Teaching Assistant
Department of Statistics & Applied Probability, University of California Santa Barbara

- Lecturer for PSTAT 501 (TA Training Course) (F16 W16)
- TA for PSTAT 213ABC (PhD Level Probability Theory) (F15 F16 W16 W17 S16 S17)
- TA for PSTAT 160A (Introduction to Stochastic Processes) (F15)
- TA for PSTAT 171 (Mathematics of Interest) (F13 F14)
- TA for PSTAT 172AB (Actuarial Statistics) (W14 W15 W16 S14 S15 S16)

- Lecturer for PSTAT 182T (Tutorial for Exam P & FM) (W14 S14)

Department of Statistics & Probability, Michigan State University

- TA for STT 315 (Introduction to Probability & Statistics for Business) (S12, S13, F13)
- TA for STT 455/456 (Actuarial Models) (F13, S13)

Honors & Awards

Recipient of SOA Hickman Scholarship (Spring 2015–Spring 2017)

- Worldwide scholarship for PhD students pursuing academia & actuarial credentials
- Only five new scholars awarded annually

Invited Lectures

Twelfth International Longevity Risk and Capital Markets Solutions Conference in Chicago (September 2016)
Topic: Gaussian Process Models for Mortality Rates and Improvement Factors

50th Actuarial Research Conference (ARC), University of Toronto (August 2015)
Topic: Statistical Emulators & Longevity Risk

Eleventh International Longevity Risk and Capital Markets Solutions Conference at Université Lyon 1, Lyon, France (September 2015)
Topic: Statistical Emulators & Longevity Risk

Seminar Talks

UCSB Statistics Department Gaussian Process Research Group (November 2016)
Newly established quarterly seminar for faculty and PhD students to discuss topics and their current research in Gaussian Processes
Topic: Stochastic Kriging in Quantile Estimation with Applications to VaR Calculations

UCSB Statistics Department Colloquium Talk (May 2016)
Topic: Statistical Emulators & Gaussian Processes

UCSB Mathematics Department (May 2015)
Topic: Proving the Central Limit Theorem in the strong operator topology

Conferences

8th Western Conference in Mathematical Finance (March 2017)
University of Washington

Society of Actuaries Annual Meeting & Exhibit (October 2015)
Austin, TX

Second NUS-UParis Diderot Workshop on Quantitative Finance (September 2015)
University of Paris Diderot

Conference on Stochastic Asymptotics & Applications (September 2014)
Joint with Sixth Western Conference on Mathematical Finance
University of California Santa Barbara

49th Actuarial Research Conference (ARC) (July 2014)
University of California Santa Barbara

Extra-Curricular Activities

Led student research group studying Continuous Martingales and Brownian Motion by Revuz & Yor (Fall 2015 – Spring 2016)

Member of SOA Education & Research Section (Summer 2016 – Current)

Member of SIAM (Spring 2012 – Current)

Abstract

Three Applications of Gaussian Process Modeling in Evaluation of Longevity Risk Management

by

James Kenneth Risk

Longevity risk, the risk associated with people living too long, is an emerging issue in financial markets. Two major factors related to this are with regards to mortality modeling and pricing of life insurance instruments. We propose use of Gaussian process regression, a technique recently populuarized in machine learning, to aid in both of these problems. In particular, we present three works using Gaussian processes in longevity risk applications. The first is related to pricing, where Gaussian processes can serve as a surrogate for conditional expectation needed for Monte Carlo simulations. Second, we investigate value-at-risk calculations in a related framework, introducing a sequential algorithm allowing Gaussian processes to search for the quantile. Lastly, we use Gaussian processes as a spatial model to model mortality rates and improvement.

Permissions and Attributions

1. The content of Chapter 4 is the result of a collaboration with Mike Ludkovski, and has previously appeared in the journal Insurance: Math and Economics (IME) Risk and Ludkovski (2016). It is reproduced here with the permission of Rob Kaas from IME.

2. The content of Chapter 6 is the result of a collaboration with Mike Ludkovski and Howard Zail Ludkovski et al. (2016). This paper has been submitted to the ASTIN Bulletin: The Journal of the International Actuarial Association. Its contents are reproduced here with the permission of Christian Levac from ASTIN Bulletin.

3. The content of Chapter 5 is the result of a collaboration with Mike Ludkovski. The paper is to be submitted in September 2017.

Preface

This work is a compilation of three completed papers, along with a unifying introduction. At the time of writing (July, 2017) one has been published in the journal Insurance: Mathematics and Economics (IME), one has been submitted to ASTIN Bulletin: The Journal of the International Actuarial Association, and one is awaiting submission. Mr. Risk is the corresponding author on two of the three papers, and in all three he developed all of the proofs (when applicable), performed all of the numerical work, and contributed to a majority of the writing.

Contents

Chapter 1

Introduction

The past few decades have seen the average life expectancy of the human population increasing more quickly than anticipated. Consequently, longevity risk, the risk associated with people living longer than expected, is becoming a rising issue. Financial markets have been impacted significantly, for example, pension issuers who neglected this increase are now paying more than initially planned for. There are a number of aspects to the longevity risk problem. First, there is a need for good mortality models that can provide faithful fits and forecasts, allowing us to measure and understand potential risks arising in the future and accurately plan for life expectancy increases. Second, products should be developed that somehow mitigate this risk, either through the payoff structure itself or by some sort of longevity swap. Third, risk managers must be able to accurately price these products and also analyze quantities such as value-at-risk for solvency requirements. Combining this task with complicated mortality models provides for a difficult problem.

The first major breakthrough in mortality modeling comes from Lee and Carter (1992), who proposes a stochastic model for mortality. As a stochastic process

depending on many factors, stochastic mortality models allow to pinpoint how sources of risk evolve in both age and time, as well as regards to other factors. In addition, the stochastic framework allows for generation of a range of future longevity forecasts, so that analysis (including pricing of mortality related instruments) can be done through Monte Carlo simulation. Since the initial work, there has been a particular interest in building new stochastic models of mortality and expanding specifically on the work of Lee and Carter (1992). Research has shown that the initial model from Lee and Carter (1992) is far from exhaustive in capturing different populations mortality dynamics, since mortality rates differ according many factors beyond age and calendar year, such as income, country, and gender. The need to produce faithful fits and forecasts has yielded a construction of increasingly complex mortality models for the mortality process $\mu(t, x)$, either as extensions of the work in Lee and Carter (1992) or from new paradigms. The latest generation of models feature multi-dimensional, nonlinear stochastic state processes driving $\mu(\cdot, x)$, see e.g. Cairns et al. (2009a); Li et al. (2009); Lin et al. (2013); Barrieu et al. (2012); Fushimi and Kogure (2014). While these models are effective at calibration and emitting reliable forecasts, they lack tractability in terms of closed-form formulas. This causes even the most basic pricing problem to be intractible, requiring numerical approximations or simulations to generate a price.

In regards to new products, one approach to mitigate risk is through the payoff structure. For example, types of variable annuities pay the buyer the maximum between a fixed rate and the growth rate on a mutual fund until time of death. Compared basic deferred annuities with a strict fixed rate, this approach allows

the seller to give a lower fixed rate due to the potential upside of mutual fund growth, see Lin and Tan (2003); Qian et al. (2010) for details. If the mutual fund does poorly, then the seller can simply give a low rate to the annuitant, and if it does well, then the seller obtains a high rate of return on the mutual fund investment. In either case, the risk associated with the annuitant living longer than expected is mitigated compared to a simple fixed rate. Still, this approach only provides partial minimization in loss due to potential longevity, since the insurer is still making payments. Alternatively, one would like a type of *mortality swap* that eliminates the risk more effectively or even altogether. In some cases, there have been transactions of complete mortality swaps customized specifically to a certain plan, see Kessler et al. (2015) for details. However, finding a buyer and further agreeing on a price is a more difficult matter, since mortality is far from liquid. Alternatives in the form of securities linked to an index of mortality do exist (see www.LLMA.org). These give obvious rise to basis risk, since there is no reason a general mortality index should reflect that of the customers of an arbitrary insurance company. Assessment of this basis risk is difficult, along with determining the risk appetites of the seller, see Cairns and El Boukfaoui (2017) for a discussion. Regardless, some residual risk may be acceptable if the price is reasonable. However, a further difficulty stems from trying to customize a swap to a specific company, due to widely varying ages, genders, economic class, etc.. In any case, there is an obvious need for accurate pricing and assessment of all of these products, though their complicated nature typically require numerical approximations and/or expensive simulation to price even under simple non-stochastic mortality models.

Clearly, the pricing of mortality-linked contracts is becoming increasingly complex, especially when working under a stochastic mortality framework. Effectively, one is required to feed multi-dimensional stochastic inputs through a "black box" that eventually yields an approximate net present value of the claim. Many emerging problems require use of *nested simulations* which can easily take days to complete. For example, to compute a one year value-at-risk through Monte Carlo, one needs to generate scenarios (e.g. mortality and market factors) one year into the future, and then for each scenario, compute the price given the scenario, which requires further simulation. Finally, one takes a quantile of the aggregate prices from each initial scenario. To combat this inevitability, practitioners generally rely on crude numerical approximations and/or inefficient simulation techniques, greatly reducing accuracy and introducing large bias potential.

This dissertation has three main advances in the problem of longevity risk analysis. The first two focus on the nested simulation problem, where we introduce the concept of *statistical emulation* to attack it. The idea of emulation is to replace the computationally expensive process of running expensive computer code $f(z)$ (for example, Monte Carlo simulations) for each scenario z with a cheap-to-evaluate surrogate model that statistically predicts $f(z)$ for any $z \in \mathbb{R}^d$ based on results from a training dataset. At the heart of emulation is statistical learning. Namely, the above predictions are based on first obtaining pathwise estimates y^n, $n = 1, \ldots, N$ of $f(z^n)$, for a set of training locations, called a design $\mathcal{D} \doteq \big((z^1, y^1), \ldots, (z^N, y^N)\big)$. Next, one regresses $\{y^n\}$ against $\{z^n\}$ to "learn" the *estimated* response surface $\hat{f}(\cdot)$. The regression aspect allows to borrow information across different scenarios starting at various sites. This reduces computational

budget compared to the nested simulation step of independently making further trajectories from *each* scenario by using the surrogate $\hat{f}$ to efficiently predict the value $f(z)$. The conceptual need for emulation is two-fold. First, the emulator is used for interpolation, i.e. using existing design to make predictions at new sites z. Second, the emulator *smoothes* any observation error, for example the byproduct of Monte Carlo simulation.

Emulation now involves the (i) experimental design step of proposing a design $\mathcal{D}$ that forms the training dataset, and (ii) a learning procedure that uses the queried results $(z^n, y^n)_{n=1}^N$, with the y^n being realizations of $f(z^n)$, to construct a fitted response surface $\hat{f}(\cdot)$. We specifically focus on Gaussian processes (GP) as emulators. As a modeling tool, GPs enjoy several advantages over other methods. The GP framework is Bayesian, offering closed form formulas for mean and standard deviation, as well as easy to simulate sample paths for pricing and other risk analysis. Additionally, it easily handles missing data, as the posterior mean and covariances can be evaluated anywhere. Similarly, with a reasonably chosen (or fitted) mean function, the GP provides informative extrapolation results, accounting for uncertainty as it predicts away from fitted data. We also mention that GPs have easy updating equations when new data is added to the model.

Chapter 4 compares various emulators for pricing. Emulators are useful here since pricing with Monte Carlo is often done using tower property, either because the modeler is given a preset collection of future scenarios to price over, or because there is some change in the model when the deferral period for an annuity ends, such as mortality refitting. In this case the design $\mathcal{D}$ should be somehow accurate globally, for all potential scenarios to be averaged over. The main conclusion is

that Gaussian processes are most practical in a basic longevity risk pricing setup. Specifically, they offer a natural framework for handling Monte Carlo noise, as well as providing a complete posterior distribution useful for risk analysis.

In Chapter 5, we use GPs as emulators for computing capital requirements, including value-at-risk (VaR) and tail value-at-risk (TVaR). We consider the typical setup in insurance solvency capital requirement calculations, which is that the modeler is given a fixed set of future scenarios, and then is required to compute a quantile (or tail average) of the net present value conditioned over each scenario. The quantile problem boils down to finding the level of a contour set. This introduces a new dimension of complexity, where one is interested in a specific scenario (or region of scenarios in the case of TVaR) instead of a desire to be accurate globally. In particular, the focus of this paper is on how to construct the design $\mathcal{D}$. The general contour problem already has some work done using Gaussian processes (Picheny et al. (2010)), but in different contexts and under different assumptions. For example, our evaluator (conditional expectation) is noisy, whereas it is not in the current literature. Further, current work assumes a known contour level and seeks the contour set itself, whereas we are interested specifically in the value of the unknown level. Thus, our contribution is coming up with methods that can attack the problem of finding an unknown level in the noisy context. In particular, we introduce sequential design algorithms that search for the tail, with nature similar to a numerical minimization or maximization problem.

The final contribution is an application of Gaussian processes to mortality modeling, discussed in Chapter 6. As mentioned, the need for accurate models is increasing. Recently, focus has been on stochastic mortality models that

are powerful, but have strong parametric assumptions along with no closed form of distribution. Consequently, they require simulations to project future scenarios. We apply the GP framework as a spatial model directly on mortality data, with the inputs being age and calendar year. In addition to being nonparamteric, Gaussian processes yield a particular benefit over other models in analyzing mortality improvement, since linear combinations (e.g. finite differences) and even their derivatives (under mild assumptions) remain Gaussian processes, with closed form for their mean and covariance functions.

We provide a brief introduction to Gaussian processes as a Bayesian model in Chapter 2, as this is the basis of the work in all of the papers. This includes the general problem framework, as well as an example illustrating the posterior properties of GPs as a modeling tool in addition to showcasing statistical emulation. Chapter 3 provides an outline of the three works and their abstracts, along with comments on future work.

Chapter 2

Introduction to Gaussian processes

As a modeling technique, the objective with Gaussian processes is to establish its Bayesian properties when observing a collection of inputs and outputs called an *design* $\mathcal{D} = \big((z^1, y^1), \ldots, (z^N, y^N)\big)$. We first introduce GPs as they are typically defined in a probability theory class.

Remark. Use of Gaussian processes as a modeling technique is also equivalently called *kriging*, originating from its initial use in geostatistics (Krige, 1951), where z represents a geographical location and y is a response, e.g temperature. This term is used mostly in place of GP in Chapter 4.

2.1 Gaussian Process Preliminaries

Let I be an arbitrary topological index set, and f an unknown function to be learned. We begin by placing a prior on f as a Gaussian process by specifying a mean and covariance structure, and use $\mathcal{D}$ to *learn* it to obtain a posterior Gaussian process $\hat{f}$.

Definition 1. *A* Gaussian process *is a collection of random variables* $(f(z^i))_{i\in I}$*, any finite number of which have a joint Gaussian distribution.*

The following framework holds for general separable metric spaces (I, d). The typical case is where $I = \mathbb{R}^+$ representing time, or $I = \mathbb{R}^d$ for modeling applications, and in either case we the usual Euclidean metric. We define the mean function $\mu(z^i)$ and covariance function $C(z^i, z^j)$ of f as

$$\begin{aligned} \mu(z^i) &= \mathbb{E}[f(z^i)] \\ C(z^i, z^j) &= \mathbb{E}[(f(z^i) - \mu(z^i))(f(z^j) - \mu(z^j)], \qquad i, j \in I. \end{aligned} \tag{2.1}$$

Generally, one first specifies a mean and covariance function and is then interested in the resulting Gaussian process. The existence and uniqueness of such a process is given by the Kolmogorov extension theorem, see Proposition 1.3.7 in Revuz and Yor (2013). In fact, it is enough to specify any symmetric semi-definite positive function (instead of covariance), as this proposition ensures it is the covariance function of a unique Gaussian process. After specifying the mean and covariance, one obtains a GP with multivariate normal finite-dimensional distributions, i.e. for any finite set $z^1, \dots, z^N$, $\boldsymbol{f} \doteq (f(z^1), \dots, f(z^N)) \in \mathbb{R}^N$ has a multivariate normal distribution with mean $\boldsymbol{\mu} \doteq (\mu(z^1), \dots, \mu(z^N))$ and covariance $\mathbf{C} \doteq [C(z^m, z^n)]_{1\le m,n\le N}$, written $\boldsymbol{f} \sim MVN(\boldsymbol{m}, \mathbf{C})$. The next objective is to construct a conditional GP given a design set $\mathcal{D} = ((z^1, y^1), \dots, (z^N, y^N))$. We assume a general setting where the outputs $y^1, \dots, y^N$ have additional noise, i.e. they are realizations of

$$Y(z) = f(z) + \epsilon(z), \tag{2.2}$$

where we identify $f(\cdot)$ as the true *response surface*, and $\epsilon(\cdot)$ is the sampling noise. We make the assumption that $\epsilon(\cdot)$ is independent and identically distributed across the domain, and that $\epsilon(z) \sim N(0, \tau^2(z))$, where $\tau^2(z)$ is the sampling variance that depends on the location z; this normality assumption along with independence implies that Y is a Gaussian process.

To determine the posterior distribution, first review the conditional multivariate normal distribution. Let $\boldsymbol{X} \sim MVN(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, and split it into two parts:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_1 \\ \boldsymbol{X}_2 \end{bmatrix}, \qquad \boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \qquad \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix}, \tag{2.3}$$

then the conditional distribution of $\boldsymbol{X}_2$ given $\boldsymbol{X}_1$ is also multivariate normal, with

$$\boldsymbol{X}_2 | \boldsymbol{X}_1 \sim MVN(\boldsymbol{\mu}_2 + \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}(\boldsymbol{X}_1 - \boldsymbol{\mu_1}), \boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}). \tag{2.4}$$

Now look at $\boldsymbol{f}$ and $\boldsymbol{y} = (y^1, \ldots, y^N)$ from Equation (2.2). Note that the equation implies $\boldsymbol{y}|\boldsymbol{f} \sim MVN(\boldsymbol{f}, \boldsymbol{\Delta})$, where $\boldsymbol{\Delta}$ is the diagonal matrix for $\epsilon(\cdot)$ with entries $\tau^2(z^1), \ldots, \tau^2(z^N)$. It also implies that

$$\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{y} \end{bmatrix} \sim MVN\left(\begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu} \end{bmatrix}, \begin{bmatrix} \mathbf{C} & \mathbf{C} \\ \mathbf{C} & \mathbf{C} + \boldsymbol{\Delta} \end{bmatrix} \right). \tag{2.5}$$

Hence, the posterior distribution of $\boldsymbol{f}$ given the design $\mathcal{D}$ (equivalent to conditioning on $\boldsymbol{y}$ since the z^n are assumed to be known) is obtained by Equation (2.4):

$$\boldsymbol{f}|\mathcal{D} \sim MVN(\boldsymbol{\mu} + \mathbf{C}(\mathbf{C}+\boldsymbol{\Delta})^{-1}(\boldsymbol{y}-\boldsymbol{\mu}), \mathbf{C}(\mathbf{C}+\boldsymbol{\Delta}^{-1})\mathbf{C}). \tag{2.6}$$

The next objective is to obtain the predictive distribution. If the test set is $\boldsymbol{z}' = (z'^1, \ldots, z'^M)$, and $\boldsymbol{f}' = (f(z'^1), \ldots, f(z'^M))$, then

$$\begin{bmatrix} \boldsymbol{f}' \\ \boldsymbol{y} \end{bmatrix} \sim MVN\left(\begin{bmatrix} \boldsymbol{\mu}' \\ \boldsymbol{\mu} \end{bmatrix}, \begin{bmatrix} \mathbf{C}_{\boldsymbol{z}'\boldsymbol{z}'} & \mathbf{C}_{\boldsymbol{z}'\boldsymbol{z}} \\ \mathbf{C}_{\boldsymbol{z}\boldsymbol{z}'} & \mathbf{C}+\boldsymbol{\Delta} \end{bmatrix} \right), \tag{2.7}$$

where $C_{\boldsymbol{z}'\boldsymbol{z}'}$ is the covariance matrix of $\boldsymbol{f}'$ over $\boldsymbol{z}'$, $\boldsymbol{C}_{\boldsymbol{z}\boldsymbol{z}'}$ is an $N \times M$ matrix with (i,j)-entry $C(z^i, z'^j)$ and $\boldsymbol{C}_{\boldsymbol{z}'\boldsymbol{z}} = \boldsymbol{C}_{\boldsymbol{z}\boldsymbol{z}'}^T$. Then the conditioning Equation (2.4) implies the posterior predictive distribution

$$\boldsymbol{f}'|\mathcal{D} \sim MVN(\boldsymbol{\mu}' + \boldsymbol{C}_{\boldsymbol{z}'\boldsymbol{z}}(\boldsymbol{C}+\boldsymbol{\Delta})^{-1}(\boldsymbol{y}-\boldsymbol{\mu}'), \boldsymbol{C}_{\boldsymbol{z}'\boldsymbol{z}'} - \boldsymbol{C}_{\boldsymbol{z}'\boldsymbol{z}}(\boldsymbol{C}+\boldsymbol{\Delta})^{-1}\boldsymbol{C}_{\boldsymbol{z}\boldsymbol{z}'} \tag{2.8}$$

Thus, we have a closed form for the predictive distribution at any set of inputs $\boldsymbol{z}'$. Note that the Kolmogorov extension theorem implies existence of a unique Gaussian process governed by these means and covariances; we define it as $\hat{f}(\cdot) \equiv \hat{f}(\cdot|\mathcal{D})$. Often used in the sequel is the case where the input is univariate, in which case we write

$$\begin{cases} m(z) \doteq \mu(z) + \mathbf{c}(z)^T(\mathbf{C}+\boldsymbol{\Delta})^{-1}(\mathbf{y}-\boldsymbol{\mu}); \\ s^2(z,z') \doteq C(z,z') - \mathbf{c}(z)^T(\mathbf{C}+\boldsymbol{\Delta})^{-1}\mathbf{c}(z'), \end{cases} \tag{2.9}$$

These predictive equations are crucial in GP modeling, and they appear in later chapters in Equations (4.23), (5.13) and (6.6), with respective modifications for their specific framework.

2.1.1 Covariance kernels

The covariance function $C(\cdot,\cdot)$ is a crucial part of a GP model. In practice, one usually considers spatially stationary or isotropic kernels,

$$C(z^i, z^j) \equiv c(z^i - z^j) = \sigma^2 \prod_{n=1}^{d} g((z^i - z^j)_n; \theta_n), \qquad \theta_n > 0 \tag{2.10}$$

reducing to the one-dimensional base kernel g, where $(z^i - z^j)_n = d_n(z^i, z^j) = d_n(0, z^i - z^j)$ is the nth coordinate distance, in this case translation invariant. The two most common choices for GP modeling are the Matern-ν kernel and Gaussian kernel

$$g_{\text{m-}\nu}(h;\theta) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{h}{\theta}\right)^{\nu} K_\nu \left(\sqrt{2\nu}\frac{h}{\theta}\right) \tag{2.11}$$

$$g_{\text{gau}}(h;\theta) = \exp\left(-\frac{h^2}{2\theta^2}\right), \qquad h \geq 0 \tag{2.12}$$

where Γ is the gamma function, K_ν is the modified Bessel function of the second kind, and ν is a non-negative parameter. Typically, one chooses $\nu = 5/2$, in which case

$$g_{\text{m-5/2}}(h;\theta) = \left(1 + \frac{\sqrt{5}h}{\theta} + \frac{5h^2}{3\theta^2}\right) \exp\left(-\frac{\sqrt{5}h}{\theta}\right). \tag{2.13}$$

The choice of covariance function determines many properties of the resulting GP, including smoothness of their sample paths. For example, GPs with Matern-

ν covariance kernels are kth order differentiable (in mean-square) if and only if $\nu > k$. GPs with Gaussian covariance kernels as in Equation (2.12) possess derivatives of all orders, and are thus very smooth (Rasmussen and Williams (2006)).

The hyper-parameters θ_j are called characteristic length-scales and can be informally viewed as roughly the distance you move in the input space before the response function can change significantly (Rasmussen and Williams, 2006, Ch 2). Constructing a GP requires picking a kernel family and the hyper-parameters σ_j, θ_j.

Section 2.5 showcases an example comparing the Matern 5/2 kernel with the Gaussian kernel. In general, the posterior equations do not change much for well behaved data, as long as the covariance kernels produce GPs with reasonably similar properties, such as smoothness of sample paths.

2.1.2 Mean Function

The case where a prior mean function μ is prespecified is also called *simple kriging*. One can also specify a parametric trend function of the form $\mu(z) = \beta_0 + \sum_{j=1}^{p} \beta_j h_j(z)$ where β_j are constants to be estimated, and $h_j(\cdot)$ are given basis functions, see Section 2.3.1 for details. This framework is called *universal kriging*, with the case $\mu(z) = \beta_0$ called *ordinary kriging*. To analyze the impact of the mean function on the posterior distribution, look at Equation (2.9). As z deviates in space from the design, the covariances decrease, so that each entry of $\mathbf{c}(z) \to 0$, resulting in the second term of $m(z)$ approaching zero. In other words, the GP reverts to its prior mean as it leaves the design space. On the other hand,

for large covariances, the second term dominates, so that the nearby trend terms end up canceling with $\mu(z)$. Thus, for prediction, the trend function has little impact in sample and is mostly important for extrapolation.

Although for prediction the mean function is mostly desired for extrapolation, a reasonably accurate mean function is wanted for hyperparameter fitting, since the θ parameters influence spatial dependence, something that detrending would affect.

2.2 Representer Theorem

One can obtain the posterior mean in Equation (2.9) by means of the *representer theorem* (Kimeldorf and Wahba (1971)) in the general theory of function regularization with reproducing kernel Hilbert spaces (RKHS). First, begin with a RKHS $\mathcal{H}$ of real functions g defined by the kernel C, i.e. for every $z, C(z, z') \in \mathcal{H}$ and $\langle g(\cdot), C(\cdot, z)\rangle_{\mathcal{H}} = g(z)$. The Moore-Aronszajn theorem ensures the RKHS is uniquely defined by C. Here, C can be any positive definite kernel, but for our purposes it is the covariance function of a GP. The representer theorem states that for the regularizing functional

$$J[g] = \frac{\lambda}{2}\|g\|^2_{\mathcal{H}} + Q(\mathbf{y}, \mathbf{g}), \tag{2.14}$$

the minimizer $g \in \mathcal{H}$ has form $g(z) = \sum_{n=1}^{N} \alpha^n C(z, z^n)$. Here, $\mathbf{y} = (y^1, \ldots, y^N)$, $\mathbf{g} = (g(z^1), \ldots, g(z^n))$, and λ is a scaling parameter trading off the two terms. The first term $\|g\|^2_{\mathcal{H}}$ is called the *regularizer* and represents smoothness assumptions on g, and $Q(\mathbf{y}, \mathbf{g})$ is a data-fit term assessing the prediction quality $g(z^n)$ for the

observed y^n. For the specific form

$$Q(\mathbf{y}, \mathbf{g}) = \frac{1}{2\tau^2} \sum_{n=1}^{N} (y^n - g(z^n))^2$$

of the negative log-likelihood of a Gaussian distribution with constant variance τ^2, we show that the minimizer of Equation (2.14) is the posterior mean of a Gaussian process with covariance kernel C. This follows by setting $g(z) = \sum_{n=1}^{N} \alpha^n C(z, z^n)$ (the solution by the representer theorem) and using the RKHS property $\langle C(\cdot, z^m), C(\cdot, z^n) \rangle_{\mathcal{H}} = C(z^m, z^n)$. Then plugging into Equation (2.14) shows

$$\begin{aligned} J[\boldsymbol{\alpha}] &= \frac{1}{2} \boldsymbol{\alpha}^T \boldsymbol{C} \boldsymbol{\alpha} + \frac{1}{2\tau^2} |\mathbf{y} - \mathbf{C}\boldsymbol{\alpha}|^2 \\ &= \frac{1}{2} \boldsymbol{\alpha}^T (\boldsymbol{C} + \frac{1}{\tau^2} \boldsymbol{C}^2) \boldsymbol{\alpha} - \frac{1}{\tau^2} \mathbf{y}^T \boldsymbol{C} \boldsymbol{\alpha} + \frac{1}{2\tau^2} \mathbf{y}^T \mathbf{y}. \end{aligned}$$

By differentiating to minimize J, the resulting minimizer for $\boldsymbol{\alpha}$ is $\boldsymbol{\alpha} = (\boldsymbol{C} + \tau^2 I)^{-1} \mathbf{y}$, so that $g(z) = \boldsymbol{c}(z)^T (\boldsymbol{C} + \tau^2 I)^{-1} \mathbf{y}$, the same as the posterior mean in Equation (2.9) with constant noise variance τ^2.

While this does not yield the full Gaussian likelihood, it does make a connection with other modeling techniques that lie in the RKHS framework, splines being one example, which are used as a comparitive emulator in Chapter 4.

2.3 Hyperparameter Estimation

The hyperparameter family depends on the mean and covariance functions, as well as the noise variances $\tau^2(z^1), \ldots \tau^2(z^N)$. For the remainder of the paper,

we consider the stationary kernels in Equation (2.10), so that the hyperparameters include σ^2 and $\theta_1, \ldots, \theta_d$. If it is reasonable to believe the noise variance is homoscedastic, then one can define the new hyperparameter $\tau^2 \equiv \tau^2(z^n)$, reducing to a single noise variance. In this case, τ^2 is called the *nugget effect*. Two common estimation methods of the hyperparameters are maximum likelihood, using the likelihood function based on the posterior distributions described through Equation (2.9), and penalized MLE. Either case leads to a nonlinear optimization problem to fit θ_j and process variance σ^2. We utilize the `R` packages `DiceKriging` Roustant et al. (2012a) and `hetGP` Binois et al. (2016) that allow fitting of kriging models for several kernel families by Maximum Likelihood.

When $\tau^2(z)$ is heteroskedastic, the estimation problem becomes more difficult. Typically, the surface $\tau^2(z)$ is estimated by some other means while the remaining hyperparameters are fitted via MLE. Binois et al. (2016) discusses a way of simultaneously modeling the surface $\tau^2(z)$ along with the GP. This is related to *stochastic kriging*, discussed in Section 2.4. An alternative way is to utilize prior knowledge, for example use of the binomial distribution as noise in the mortality application is discussed near the end of Section 6.3.2.

2.3.1 Mean Function Estimation

One can specify a parametric trend function of the form $\mu(z) = \beta_0 + \sum_{j=1}^{p} \beta_j h_j(z)$ where β_j are constants to be estimated, and $h_j(\cdot)$ are given basis functions. In this case, the coefficient vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is estimated simultaneously with the Gaussian process component $\hat{f}(\cdot)$. Basis function choice is determined either through prior knowledge about f, or through data visualization.

Define $\mathbf{h}(z) \doteq (h_1(z), \ldots, h_p(z))$ and $\mathbf{H} \doteq \left(\mathbf{h}(z^1), \ldots, \mathbf{h}(z^N)\right)$, then posterior mean and variance at location z with estimated trend function included are Roustant et al. (2012a)

$$\begin{cases} m_h(z) = \mathbf{h}(z)^T\hat{\boldsymbol{\beta}} + \mathbf{c}(z)^T\mathbf{C}^{-1}(\mathbf{y} - \mathbf{H}\hat{\boldsymbol{\beta}}); \\ s_h^2(z) = s^2(z) + \left(\mathbf{h}(z)^T - \mathbf{c}(z)^T\mathbf{C}^{-1}\mathbf{H}\right)^T \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1} \left(\mathbf{h}(z)^T - \mathbf{c}(z)^T\mathbf{C}^{-1}\mathbf{H}\right), \end{cases} \tag{2.15}$$

where the best linear estimator of the trend coefficients $\boldsymbol{\beta}$ is given by the usual linear regression formula $\hat{\boldsymbol{\beta}} \doteq \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{C}^{-1}\mathbf{y}$.

The combination of trend and GPs offers an attractive framework for fitting a response surface. The trend component allows to incorporate domain knowledge about the response, while the GP component offers a flexible nonparametric correction.

Remark. The case where a prior μ is specified is called *simple kriging*; when it is estimated by means of this section it is named *universal kriging*, with the subcase $\mu(z) = \beta_0$ called *ordinary kriging*.

2.3.2 Bayesian GP and MCMC

One can also consider a fully Bayesian GP model, where the mean and/or covariance parameters have a prior distribution, see Williams and Rasmussen (2006). Bayesian GP implies that there is additional, intrinsic uncertainty about C which is propagated through to the predictive distributions $\hat{f}$. Starting from the hyper-prior $p(\Theta)$, the posterior distribution of the hyperparameters is obtained

via $p(\Theta|\mathcal{D}) \propto p(\Theta)p(\mathbf{y}|\mathbf{z},\Theta)$. This hierarchical posterior distribution is typically not a GP itself. Practically this means that one draws realizations Θ^m, $m = 1, 2, \ldots$ from the posterior hyperparameters and then applies (2.9) to each draw to compute $m(z|\Theta^m), C(z, z'|\Theta^m)$.

In general, sampling from $p(\Theta|\mathcal{D})$ requires approximate techniques such as Markov Chain Monte Carlo. The output of MCMC is a sequence $\Theta^1, \Theta^2, \ldots, \Theta^M$ of Θ values which can be used as an empirical approximation for the marginal distribution of Θ, namely $p(\Theta|\mathbf{y}, \mathbf{z})$. From this sequence, it possible to calculate means and modes of the model parameters or use the Θ sequence directly to conduct posterior predictive inference. A hybrid approach first specifies hyperparameter priors but then simply uses the MAP estimates of Θ for prediction (thus bypassing the computationally intensive MCMC steps). This idea is motivated by the observation that under a vague prior $p(\Theta) \propto 1$, the posterior of Θ is proportional to the likelihood, so that the MAP estimator $\hat{\Theta}$ which optimizes $p(\Theta|\mathbf{y}, \mathbf{z})$ becomes identical to the MLE maximizer above.

We note that standard MCMC techniques are not well suited for GP as the components of Θ tend to be highly correlated resulting in slow convergence of the MCMC chains. One solution is to use Hamiltonian Monte Carlo (HMC) (Brooks et al., 2011) which is better equipped for managing correlated parameters.

It is also possible to attach a prior to the mean function and/or covariance structure itself. This has recently been done in functional data analysis: Yang et al. (2017) and Yang et al. (2016) apply an Inverse-Wishart Process (IWP) to the covariance function, and a separate GP to the mean function (sharing the same IWP covariance prior). The IWP has marginal Inverse-Wishart distributions, a

distribution family over matrices. Fitting is done using an MCMC algorithm; see Yang et al. (2016) for details. The flexibility of this approach has the tradeoff of being relatively expensive compared to non-Bayesian techniques. Yang et al. (2017) proposes a numerical speed-up for when the number of grid points N is large.

2.4 Emulation and Stochastic Kriging

Recall that the goal in emulation is to learn an unknown function f by evaluating it at various points and using these observations to predict nearby. GPs work well in this framework since they are driven by spatial correlation as seen in the posterior Equations (2.9). Often in emulation, the observations are noisy ($\tau^2 > 0$), for example in Monte Carlo simulations. In this case, one usually repeats observations, i.e. Equation (2.2) is replicated r^n for the same location z^n to obtain $y^{n,1}, \ldots, y^{n,r^n}$. Here, the *output* for the design at location z^n is $\bar{y}^n = \sum_{i=1}^{r^n} y^{n,i}$ with (Monte Carlo) observation noise $\tau^2(z^n)/r^n$. Under this framework, use of the GP posterior equations (2.9) is called *stochastic kriging* (Ankenman et al. (2010)). Note that in the typical case of $\tau^2(z)$ being unknown and heteroscedastic, one can now estimate it by a sample variance estimator

$$\hat{\tau}^2(z^n) = \frac{1}{r^n - 1} \sum_{i=1}^{r^n} (y^{n,i} - \bar{y}^n)^2. \tag{2.16}$$

When using $\hat{\tau}^2(z^n)$ in replace of $\tau^2(z^n)$, the resulting posterior mean in Equation (2.9) is still unbiased, as long as $r^n > 1$, as shown in Ankenman et al. (2010) (they recommend $r^n \geq 10$). Still, this has the disadvantage of $\tau^2(z)$ being unable to be

estimated at locations where Equation (2.2) has not been ran and for low values of r^n. In the same paper, Ankenman et al. (2010) propose a separately fitted GP for the surface τ^2 trained on the pairs $(z^n, \hat{\tau}^2(z^n))_{n=1}^N$ to predict and smooth a surface for $\tau^2(z)$.

Binois et al. (2016) provides an improvement over this technique, providing an estimated noise surface based simply on the total collection of outputs $((y^{n,i})_{i=1}^{r^n})_{n=1}^N$ together with their inputs $(z^n)_{n=1}^N$. In addition to providing noise predictions at unknown locations, this also has the advantage of not requiring $r^n > 1$, since it directly links together the collection of outputs rather than modeling simply from sample variance point estimates. GP modeling with this is available in the `R` package `hetGP` Binois et al. (2016).

2.5 Emulation Example with GPs

Suppose that we have an unknown function $f(x) = \sin(x), x \in [0, 6]$. Given a fixed grid $\mathcal{Z} = (0.2, 0.4, \ldots, 5.6, 5.8)$, suppose we can obtain realizations of the noisy process $Y(z)$ in Equation (2.2), where $\epsilon(z) \sim N(0, z/10)$, so that $\tau^2(z) = z/10$. Note that the scenario set size is $|\mathcal{Z}| = 29$. Let us try to answer a few questions:

1. What does f look like?

2. What is $f(2\pi)$ (out of the design space)? How certain is the answer?

3. Where does f attain its minimum out of all points on the grid?

We answer these questions by simulating $Y(z)$ and fitting a GP $\hat{f}$ to the design $\mathcal{D} = ((z^1, y^1), \ldots, (z^N, y^N))$. For simplicity, we assume $\tau^2(z)$ is known, and that we simulate by taking $r = 1$ realization of $Y(z)$ for each $z \in \mathcal{Z}$ with fitted mean function $\mu(z) = \beta_0$. We use both the Matern 5/2 and Gaussian kernels for comparison. To determine what the true $f(\cdot)$ looks like, the GP offers two approaches: (i) the posterior mean $m(z)$ of $\hat{f}(z)$ from Equation (2.9) is also the posterior mode, so it provides the most likely function of what $f(\cdot)$ should look like, or (ii) one can simulate the GP to obtain sample realizations of $\hat{f}(z)$. Similarly, we can estimate $f(2\pi)$ by $m(2\pi)$, the posterior mean of $\hat{f}(2\pi)$ and obtain a confidence interval using the posterior standard deviation $s(2\pi)$ from Equation (2.9). This all is illustrated in Figure 2.1. The confidence bands can be obtained through the typical formula for a normal random variable, $m(z) \pm 1.96s(z)$. The resulting Matern GP has hyperparameter estimates of $\hat{\beta}_0 = -0.0245, \hat{\theta} = 1.6770$ and $\hat{\sigma}^2 = 0.5451$, and the Gaussian kernel GP has $\hat{\beta}_0 = -0.0326, \hat{\theta} = 1.4841$ and $\hat{\sigma}^2 = 0.5656$. In general, Gaussian kernel lengthscales are less than that of Matern ones because of how the kernel is parameterized.

First, note that the kernel has very little impact on the posterior mean and standard deviation; we see the Matern kernel has slightly wider confidence bands on the left side, but otherwise the posterior distribution is nearly identical between the two. We do, however, observe the Gaussian realizations are smoother than the Matern ones, as mentioned in Section 2.1.1. To answer the questions, we see that despite heavy influence from the noise, the posterior mean reasonably matches $f(z) = \sin(z)$ in shape, and that the confidence bands contain the true function almost completely. Also note that the widening confidence bands reflect

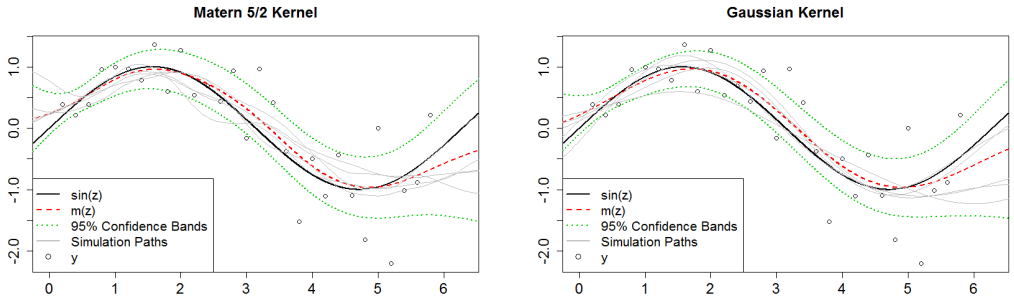

Figure 2.1: Plots of the realizations y from $Y(z), z \in \mathcal{Z}$ and posterior mean and standard deviation functions of the resulting GP $\hat{f}$, as well as simulated realizations of $\hat{f}$. Compared are the Matern 5/2 and Gaussian covariance kernels.

the increasing value of $\tau^2(z)$. We obtain for the Matern kernel $m(2\pi) = -0.4528$ and $s(2\pi) = 0.4965$ (for the Gaussian kernel, $m(2\pi) = -0.458$ and $s(2\pi) = 0.4812$). The high posterior standard deviation is due to $\tau^2(z)$ being relatively large at the right end of the grid. The figure shows the mean function declining at the right edge, this is due to the intercept only mean function $\mu(z) = \beta_0$. As GPs revert to their mean function asymptotically, they are not useful for deep extrapolation. Regardless, the uncertainty is reflected in the large value of $s(2\pi)$, so that the true value is contained within one standard deviation of the posterior mean.

To answer the third question, the minimum of $m(z), z \in \mathcal{Z}$ (i.e. over the grid) is attained at $z = 5$, while the true minimum of $\sin(z), z \in \mathcal{Z}$ is at $z = 4.6$. To better motivate the design aspect of emulation (and Chapter 5), let us investigate a way to more accurately determine the minimum. Given a remaining budget of $N_{tot} = 1000$ replications, what is the optimal way to allocate them among $z \in \mathcal{Z}$ to best find the minimum? Here, a naive uniform spreading among all $z \in \mathcal{Z}$ is ill

advised, since $\hat{f}(z)$ already has an idea of where the minimum lies, combined with uncertainty estimates. This problem is analyzed in Liu and Staum (2010) and Picheny et al. (2010); these approaches along with other sophisticated methods are discussed in Chapter 5. To provide a quick intuitive approach, consider the simple design algorithm consisting of $K = 250$ rounds that, at each step, allocates $r = 1$ replication to the $z \in \mathcal{Z}$ yielding the smallest four values of $m(z)$. The idea behind this scheme is to reduce uncertainty by adding replications only at the important area, but to also search beyond what the current guess for the minimum is, in case it is incorrect. The spatial nature of the GP ensures that no information from the simulations are wasted. Denote k as the current round of the procedure, and attach subscripts $m_k(z)$ and $s_k(z)$ for the GP mean and standard deviation for round k, with the convention that $k = 0$ means after the initialization done above. Figure 2.2 shows the analogue of Figure 2.1 for the Matern kernel when $k = 25$ and $k = 250$. Note that Figure 2.1 compares this with $k = 0$.

The obvious change is tightening of the confidence bands around the minimum region. There is obvious convergence occuring, with $m_0(4.6) = -0.9258, m_{25}(4.6) = -0.9675, m_{250}(4.6) = -0.9734$, and $s_0(4.6) = 0.2183, s_{25}(4.6) = 0.07772, s_{250}(4.6) = 0.02799$. Surprisingly, after $k = 25$ rounds, the GP correctly identified the minimum as $z = 4.6$, but at $k = 250$ the value $m_{25}(4.8) = -0.9895$ is smaller. Still, the GP will asymptotically correct itself as $N_{tot} \to \infty$.

Despite performing well, this algorithm was designed simply from intuition and has many flaws. For example, it will fail if it gets stuck on a local minimum for a non-monotonic function. Additionally, there is no reason 4 is the correct number

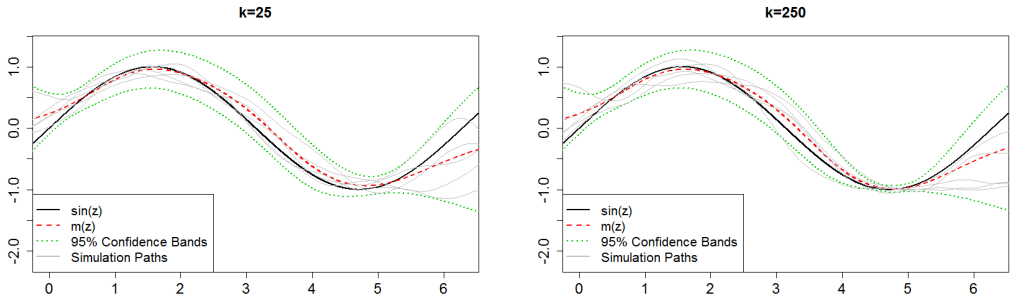

Figure 2.2: Plots of the posterior mean and standard deviation functions of the resulting GP $\hat{f}$ (with Matern 5/2 covariance kernel), as well as simulated realizations of $\hat{f}$, after $k = 25$ and $k = 250$ rounds of the sequential method attaching four replications to the locations with lowest posterior mean. The result for $k = 0$ is in Figure 2.1.

to choose; a more optimal value could be calculated based on some criteria such as MSE reducing. Specifically, a criteria that weights uncertainty as well as closeness to a minimum is attractive; this could even eliminate the issue of getting stuck at local minima.

To connect this example to the remainder of the paper, Chapter 5 focuses specifically on this problem for the case where $|\mathcal{Z}| = 10^4$ (here it was $|\mathcal{Z}| = 29$) and $z \in \mathbb{R}^d, d > 1$; specifically, we analyze two case studies where $d = 2$ and $d = 6$. In that chapter, the quantity of interest is the $\alpha-$quantile and $\alpha-$tail average rather than minimum. It discusses more deeply the algortihms involved, for example, Section 5.4.1 introduces an algorithm that sequentially chooses points to reduce a weighted mean squared error criterion. This example can also be related to how pricing is handled in Chapter 4: suppose a claim has payoff $f(U) = \sin(U)$ where $U \sim$uniform$(0, 6)$, but that f is computationally expensive to evaluate. Pricing through Monte Carlo averages $f(U)$ over simulated realizations of U. Through

the emulation framework, the resulting GP $\hat{f}$ can be used in place of f, lessening the computaitonal burden. For example, for a fixed time budget, one can produce many more realizations of U than under the original setup where it is evaluated under the expensive f, reducing Monte Carlo variance.

Chapter 3

Outline

We now present the three papers. The first is Statistical Emulators for Pricing and Hedging Longevity Risk Products, given in Chapter 4, joint work with Mike Ludkovski, published in IME in May 2016 (Risk and Ludkovski (2016)). We propose the use of statistical emulators for the purpose of analyzing mortality-linked contracts in stochastic mortality models. Such models typically require (nested) evaluation of expected values of nonlinear functionals of multi-dimensional stochastic processes. Except in the simplest cases, no closed-form expressions are available, necessitating numerical approximation. To complement various analytic approximations, we advocate the use of modern statistical tools from machine learning to generate a flexible, non-parametric surrogate for the true mappings. This method allows performance guarantees regarding approximation accuracy and removes the need for nested simulation. We illustrate our approach with case studies involving (i) a Lee-Carter model with mortality shocks; (ii) index-based static hedging with longevity basis risk; (iii) a Cairns-Blake-Dowd stochastic survival probability model; (iv) variable annuities under stochastic interest rate and mortality.

The second is Sequential Design Algorithms for Estimating Value-At-Risk for

Longevity Risk, given in Chapter 5, which is joint work with Mike Ludkovski soon to be submitted for publication. Motivated by VaR calculations for longevity risk, the work is similar to Risk and Ludkovski (2016) with the focus being on the design itself. In particular, the problem of efficient level set estimation in noisy computer experiments is largely unexplored. This problem arises often in the case of nested Monte Carlo simulation, an important example being capital requirement calculations for banking and insurance. Practitioners are forced to rely on crude numerical approximations and/or inefficient simulation techniques, greatly reducing accuracy and introducing large bias potential. The goal of this paper is to analyze methods that efficiently estimate the level of a level set to arbitrary degrees of accuracy. Using Gaussian Process (GP) regression, the end result offers rich uncertainty quantification. In particular, we improve upon several existing level set estimation techniques in the literature that suffer in various regimes of the noisy case. We also extend these methods to the case of analyzing the tail average of a contour. Our improvements are compared in two case studies along with existing techniques and other benchmarks in two case studies. The first is a two-dimensional example involving call options, and the second a six-dimensional example of longevity risk for a life annuity under stochastic interest rate and mortality.

Lastly, we present Gaussian Process Models for Mortality Rates and Improvement Factors, given in Chapter 6, which is joint work with Mike Ludkovski and Howard Zail, and was submitted to ASTIN Bulletin in April 2017. We develop a Gaussian process framework for modeling mortality rates and mortality improvement factors. GP regression is a nonparametric, data-driven approach for

determining the spatial dependence in mortality rates and jointly smoothing raw rates across dimensions, such as calendar year and age. The GP model quantifies uncertainty associated with smoothed historical experience and generates full stochastic trajectories for out-of-sample forecasts. Our framework is well suited for updating projections when newly available data arrives, and for dealing with "edge" issues where credibility is lower. We present a detailed analysis of Gaussian process model performance for US mortality experience based on the CDC datasets. We investigate the interaction between mean and residual modeling, Bayesian and non-Bayesian GP methodologies, accuracy of in-sample and out-of-sample forecasting, and stability of model parameters. We also document the general decline, along with strong age-dependency, in mortality improvement factors over the past few years, contrasting our findings with the Society of Actuaries ("SOA") MP-2014 and -2015 models that do not fully reflect these recent trends.

3.1 Notation

A best attempt was made to unify the notation in the three papers. Due to the different nature of the articles, there are some cases where this was not possible. For the most part, the notation is intuitive and self-explanatory. Regardless, we mention here a few particular differences:

1. Chapter 4 uses N_{tr} as the total simulation budget, with $N_{tr,1}$ being the design size and $N_{tr,2}$ being the (fixed) simulation budget per scenario in the design. Chapter 5 uses N_{tot} as the total simulation budget, with N being the design size and r^n being the number of *replications* for each scenario

$z^n \in \mathcal{Z}$.

2. Chapter 4 uses a superscript for indexing, e.g. $z^{(n)}$ instead of z^n which the other two chapters use. Chapter 5 uses the parentheses notation to indicate order statistics.

3. Chapter 5 uses r for replications, and β for interest rate; Chapter 4 uses r as interest rate, and along with Chapter 6 it uses β as parameters in mean functions.

4. Chapter 4 uses parentheses to denote stochastic process indexing, e.g. $Z(t)$ versus Z_t; the others use the subscript.

5. Chapter 6 uses μ as the mortality rate; the others use it as the mean function.

6. Chapter 6 presents the material in a slightly different setting, often vectorizing things through boldface.

7. Chapter 6 uses x as the input, the other chapters use z.

8. Chapter 6 uses i, j as indexing for summations and sequences, the other chapters use m, n.

3.2 Outlook

Moving forward, there are many directions for future research:

- Level estimation in a noisy setting is still greatly untouched, as our work is the first major paper analyzing it. Accurate noise estimation is crucial in

this setting, as it is the key driver in predictive uncertainty. A clear path moving forward is to find new acquisition functions based on noise surface modeling in the new work Binois et al. (2016).

- GP applications have a natural fit in numerical problems involving backward stochastic differential equations (BSDEs). This can be related to optimal stopping (Gramacy and Ludkovski (2015)) (using relation between optimal stopping and reflected BSDEs) or to general numerical problems. For general numerical problems, the most referenced paper is given by Gobet et al. (2005), "A regression-based Monte Carlo method to solve backward stochastic differential equations." Both terms "regression-based" and "Monte Carlo" invite GPs to make an entrance.

- The mortality paper has several potential extensions. For example, we can build on our current work by considering multiple populations, borrowing information from other countries and genders to unify age and period effects, simultaneously modeling static effects from the individual countries. In addition, a common case in industry arises in Cairns et al. (2011b) who use a Bayesian model to model two populations where one is a subpopulation (e.g. general public versus insured individuals). A separate direction to analyze mortality by cause, e.g. cancer versus heart disease versus accidental death. Here, there is an obvious relation of these factors with time, and breaking mortality by cause will not only produce more detailed analysis, but also provide insight into issues in the original model's assumptions.

- Mortality improvement is of crucial importance in longevity risk and is

defined as the change in mortality in time. In the current paper, the GP derivative is only briefly mentioned, and it has several elegant qualities. The second derivative for example is related to mortality acceleration, i.e. how quickly improvement factors change. This is extremely important in longevity risk analysis, it and was barely touched upon in the paper. A related idea is to add the constraint of monotonicity to the model, a quality that should hold in age. Since the GP and its derivative are jointly Gaussian, this is easily handled by conditioning on the age derivative to be positive.

Chapter 4

Statistical Emulators for Pricing and Hedging Longevity Risk Products

4.1 Introduction

Longevity risk has emerged as a key research topic in the past two decades. Since the seminal work of Lee and Carter (1992) there has been a particular interest in building stochastic models of mortality. Stochastic mortality allows for generation of a range of future longevity forecasts, and permits the modeler to pinpoint sources of randomness, so as to better quantify respective risk. Longevity modeling calls for a marriage between the statistical problem of calibration, i.e. fitting to past mortality data, and the financial problem of pricing and hedging future longevity risk. At its core, the latter problem reduces to computing expected values of certain functionals of the underlying stochastic processes. For example, the survival probability for t years for an individual currently aged x can be expressed

as a functional

$$P(t,x) = \mathbb{E}\left[\exp\left(-\int_0^t \mu(s, x+s)\, ds\right)\right], \tag{4.1}$$

where $\mu(s, x+s)$ is the force of mortality at date s for an individual aged $x+s$. In the stochastic mortality paradigm $\mu(s, x+s)$ is random for $s > 0$, and so one is necessarily confronted with the need to evaluate the corresponding expectations on the right-hand-side of (4.1).

The past decade has witnessed a strong trend towards complexity in both components of (4.1). On the one hand, driven by the desire to provide faithful fits (and forecasts) to existing mortality data, increasingly complex mortality models for $\mu(t,x)$ have been proposed. The latest generation of models feature multi-dimensional, nonlinear stochastic state processes driving $\mu(\cdot, x)$, see e.g. Cairns et al. (2009a); Li et al. (2009); Lin et al. (2013); Barrieu et al. (2012); Fushimi and Kogure (2014). These models are effective at calibration and emitting informative forecasts, but lack tractability in terms of closed-form formulas. On the other hand, sophisticated insurance products, such as variable annuities or longevity swap derivatives make valuation and hedging highly nontrivial, and typically call for numerical approaches, as closed-form formulas are not available, see e.g. Bacinello et al. (2011); Qian et al. (2010). Taken together, pricing of mortality-linked contracts becomes a complex system, feeding multi-dimensional stochastic inputs through a "black box" that eventually outputs net present value of the claim.

These developments have created a tension between the complexity of mor-

tality models that do not admit explicit computations and the need to price, hedge and risk manage complicated contracts based on such models. Due to this challenge, there remains a gap between the academic mortality modeling and the implemented models by the longevity risk practitioners. Because the aforementioned valuation black box is analytically intractable, there is a growing reliance on Monte Carlo simulation tools, which in turn is accompanied by exploding computational needs. For example, many emerging problems require *nested simulations* which can easily take days to complete. Similarly, many portfolios contain millions of heterogeneous products (see, e.g. Gan and Lin (2015)) that must be accurately priced and managed. In this article we propose to apply modern statistical methods to address this issue. Our approach is to bridge between the mortality modeling and the desired pricing/hedging needs through an intermediate *statistical emulator*. The emulator provides a computationally efficient, high-fidelity surrogate to the actual mortality model. Moreover, the emulator converts a calibrated opaque mortality model into a user-friendly valuation "app". The resulting toolbox allows a plug-and-play strategy, so that the end user who is in charge of pricing/risk-management can straightforwardly swap one mortality model for another, or one set of mortality parameters for an alternative. This modular approach allows a flexible solution to robustify the model-based longevity risk by facilitating comparisons of different longevity dynamics and different assumptions.

Use of *emulators* is a natural solution to handle complex underlying stochastic simulators and has become commonplace in the simulation and machine learning communities Santner et al. (2003); Kleijnen (2007). Below we propose to apply

such statistical learning within the novel context of insurance applications. In contrast to traditional (generalized) linear models, emulation calls for fully non-parametric models, which are less familiar to actuaries. To fix ideas, in this article we pursue the problem of pricing/hedging vanilla life annuities, a foundational task in life insurance and pension plan management. Except in the simplest settings, there are no explicit formulas for annuity values and consequently approximation techniques are already commonplace. Looking more broadly, our method would also be pertinent for computing risk measures, such as Expected Shortfall for longevity products, and in other actuarial contexts, see Section 4.8.

The paper is organized as follows: In Section 4.2 we introduce the emulation problem and review the mathematical framework of stochastic mortality. Section 4.3 discusses the construction of emulators, including spline and kriging surrogates, as well as generation of training designs and simulation budgeting. The second half of the paper then presents four extended case studies on several stochastic mortality models that have been put forth in the literature. In Section 4.4 we examine a Lee-Carter model with mortality shocks that was proposed by Chen and Cox Chen and Cox (2009); Section 4.5 studies approximation of hedge portfolio values in a two-population model based on the recent work by Cairns et al Cairns et al. (2014). Section 4.6 considers valuation of deferred annuities under a Cairns-Blake-Dowd (CBD) Cairns et al. (2006) mortality framework. Lastly, in Section 4.7 we consider variable annuities and their future distributions for risk measure analysis, using stochastic interest rate and the Lee-Carter framework.

4.2 Emulation Objective

We consider a stochastic system with Markov state process $Z = (Z(t))$. Throughout the paper we will identify Z with the underlying stochastic *mortality factors*. In Section 4.2.2 we review some of the existing such models and explicit the respective structure of Z. Typically, Z is a multivariate stochastic process based on either a stochastic differential equation or time-series ARIMA frameworks. For example, Z may be of diffusion-type or an auto-regressive process.

In the inference step, the dynamics of Z are calibrated to past mortality data that reflect as closely as possible the population of interest. In the ensuing valuation step, the modeler seeks to evaluate certain quantities related to a functional $F(T, Z(\cdot))$ looking into the future. Here F maps the stochastic factors to the present value of a life insurance product at a future date T, or alternatively the actuarially fair value of a deferred contract, common in longevity risk, that starts at T. Our notation furthermore indicates that F potentially depends on the whole path $\{Z(t), t \geq T\}$, such as

$$F(T, Z(\cdot)) = \exp\Big(-\sum_{t=T}^{\infty} h(Z(t))\Big), \tag{4.2}$$

for some $h(z)$. Given F, a common aim is to compute its expected value based on the initial data at $t = 0$,

$$\mathbb{E}\left[F(T, Z(\cdot)) \mid Z(0)\right]. \tag{4.3}$$

Another key problem is to evaluate the quantile $q(\alpha; F(T, Z(\cdot)))$, eg. the Value-at-Risk at level α of F. Other quantities of interest in actuarial applications include the Expected Shortfall of F, $\mathbb{E}[F(T, Z(\cdot)) \mid F(T, Z(\cdot)) \leq q(\alpha; F(T, Z(\cdot))), Z(0)]$ and the correlation between two functionals, $Corr(F_1(T, Z(\cdot)), F_2(T, Z(\cdot))|Z(0))$.

Our initial focus is on (4.3) which is a fundamental quantity in pricing/hedging problems. When $T > 0$, the evaluation of (4.3) can be broken into two steps, namely first we evaluate

$$f(z) \doteq \mathbb{E}[F(T, Z(\cdot))|Z(T) = z], \tag{4.4}$$

and then use the Markov property of Z to carry out an outer average,

$$\mathbb{E}[F(T, Z(\cdot))|Z(0)] = \int_{\mathbb{R}^d} f(z) p_T(z|Z(0)) dz,$$

where $p_T(z'|z) = \mathbb{P}(Z(T) = z'|Z(0) = z)$ is the transition density of Z over $[0, T]$. In addition to computing expected values from point of view of $t = 0$, computation of $f(z)$ is also necessary for analyzing the distribution of future loss in terms of underlying risk factors, e.g. for risk measurement purposes.

Crucially, because the form of $F(T, Z(\cdot))$ is nontrivial, we shall assume that $f(z)$ is not available explicitly, and there is no simple way to describe its functional form. However, since $f(z)$ is a conditional expectation, it can be sampled using a simulator, i.e. the modeler has access to an engine that can generate independent, identically distributed samples $F(T, Z^{(n)}(\cdot))$, $n = 1, \ldots,$ given $Z(0)$. However this simulator is assumed to be expensive, implying that computational efficiency is desired in using it.

Given an initial state $Z(0)$, a naive Monte Carlo approach to evaluate (4.3) is based on nested simulation. First, the outer integral over $p_T(z|Z(0))$ is replaced by an empirical average of (4.4) across $m = 1, \ldots, N_{out}$ draws $z^{(m)} \sim Z(T)|Z(0)$,

$$\mathbb{E}[F(T, Z(\cdot))|Z(0)] \simeq \frac{1}{N_{out}} \sum_{m=1}^{N_{out}} f(z^{(m)}). \tag{4.5}$$

Second, for each $z^{(m)}$ the corresponding inner expected value $f(z^{(m)})$ is further approximated via

$$f(z^{(m)}) \simeq \frac{1}{N_{in}} \sum_{n=1}^{N_{in}} F(T, z^{(m),n}(\cdot)), \qquad m = 1, \ldots, N_{out}, \tag{4.6}$$

where $z^{(m),n}(t), t \geq T$ are N_{in} independent trajectories of Z with a fixed starting point $z^{(m),n}(T) = z^{(m)}$. This nested approach offers an unbiased but expensive estimate. Indeed, the total simulation budget is $\mathcal{O}(N_{out} \cdot N_{in})$ (where the usual big-Oh notation $h(x) = \mathcal{O}(x)$ means that $h(\cdot)$ is asymptotically linear in x as $x \to \infty$) which can be computationally intensive – for example a budget of 1,000 at each sub-step requires 10^6 total simulations. As stochastic mortality models become more complex, models with $d = 3, 4, 5+$ factors are frequently proposed, and efficiency issues become central to the ability of evaluating (4.3) tractably.

For this reason, it is desirable to construct more frugal schemes for approximating (4.3). The main idea is to replace the inner step of repeatedly evaluating $f(z)$ (possibly for some very similar values of z) with a simpler alternative. One strategy is to construct deterministic approximations to (4.4) by replacing the random variable $Z(s)|Z(T)$, $s > T$ with a fixed constant, e.g. its mean, which can then be plugged into F to estimate the latter's expected value. This effectively

removes the stochastic aspect and allows to obtain explicit approximations to $f(\cdot)$. (The simplest approximation is to simply freeze $Z(s) = Z(T)\ \forall s > T$.) In some cases one can also generate upper and/or lower bounds on $f(z)$ which is helpful for risk management. However, in general the quality of an analytic formula is hard to judge, and moreover, analytic, off-line derivations are needed to obtain a good approximation. As an alternative, we therefore advocate the statistical method of utilizing a *surrogate* model for $f(\cdot)$. This approach can be generically used in any Markovian setting, requires no analytic derivations, and makes minimal a priori assumptions about the structure of $f(\cdot)$.

An emulation framework generates a fitted $\hat{f}(\cdot)$ by solving regression equations over a training dataset $\{z^{(n)}, F(T, z^{(n)}(\cdot))\}_{n=1}^{N_{tr}}$ of size N_{tr}. Emulation reduces approximating $f(\cdot)$ to the twin statistical problems of (i) experimental design (generating the training dataset) and (ii) regression (specifying the optimization problem that the approximation $\hat{f}$ solves). Details of these steps are presented in Section 4.3 below.

Because we are fitting a full response model, rather than a pointwise estimate, the emulator budget $N_{tr} \gg N_{in}$ will be an order of magnitude bigger than in (4.6). It will also require regression overhead. However, once $\hat{f}$ is fitted, prediction of $\hat{f}(z)$ for a particular value z takes $\mathcal{O}(1)$ effort, so that we can use (4.5) to estimate the original problem in (4.3) at a cost linear in N_{out}. To sum up, the total budget of the emulator is just $\mathcal{O}(N_{tr} + N_{out})$, much smaller than $\mathcal{O}(N_{out} \times N_{in})$ of nested Monte Carlo. These savings become even more significant as the dimension of state Z grows. Indeed, with multi-dimensional models, both N_{out} and N_{in} need to be larger to better cover the respective integrals over $\mathbb{R}^d$, and hence the efficiency of

nested simulations will deteriorate quickly. Intuitively, the latter computational budget is at least quadratic in d. In contrast, the intuitive complexity of an emulator is linear in d.

4.2.1 Valuation of Life Annuities

In longevity modeling, Z represents the stochastic factors driving the central force of mortality $m(t,x)$. Formally, $Z = (Z(t)) = (Z_1(t), \ldots, Z_d(t))$ is a $d-$dimensional $(\mathcal{F}(t))$ measurable Markov process on a complete filtered probability space $(\Omega, \mathcal{F}, \mathbb{P}, (\mathcal{F}(t)))$. The filtration $(\mathcal{F}(t))_{t=0}^{\infty}$ is the information up to time t of the evolution of the mortality processes.

A typical state-of-the-art model decomposes $m(t,x)$ into a longevity trend, an Age effect, and a Cohort effect (known collectively as APC models). Each of the above may be modeled in turn by one or more stochastic factors. The most common models are the Lee-Carter Lee and Carter (1992) and CBD Cairns et al. (2006) models and their generalizations. Generally their individual components follow an ARIMA model; details can be found in the survey Cairns et al. (2009a).

To deal with cashflows at different dates, we assume the existence of a risk-free asset and denote by $B(T, T+s)$ the price of an s-bond at date T with maturity at $T+s$. For the rest of the article we will assume constant force of interest r, leading to $B(T, T+s) = e^{-rs}$. As we show in Section 4.7, one can straightforwardly handle stochastic interest rates as part of $Z(\cdot)$; see also Jalen and Mamon (2009) for a discussion of correlation structure between mortality and interest rates and Fushimi and Kogure (2014) for an example that applies Bayesian methods to longevity derivative pricing under a Cox-Ingersoll-Ross (CIR) interest rate model.

Consider an individual aged x at time 0 whose remaining lifetime random variable is denoted as τ_x. The state process Z captures $m(t, x+t)$, the mortality rate process for τ_x at time t, when the individual would have aged to $x+t$. For small dt, the instantaneous probability of death is approximately $m(t, x+t)dt$, so that the random survival function of τ_x is

$$S(t,x) \doteq \exp\left(-\int_0^t m(s, x+s)\right). \tag{4.7}$$

More generally for $u \leq t < T$, the probability of an individual aged x to survive between dates t and T, given the information at time u is given by

$$\begin{aligned}\mathbb{P}(\tau_x > T \mid \tau_x > t, \mathcal{F}_u) &= \mathbb{E}\left[\frac{S(T,x)}{S(t,x)}\middle|\mathcal{F}_u\right] \\ &= \mathbb{E}\left[\exp\left(-\int_t^T m(s, x+s)\right)\middle| Z(u)\right] \doteq P(Z(u); t, T, x),\end{aligned} \tag{4.8}$$

where the last equality follows from the Markov property. The deterministic analogue of $P(Z(0); t, T, x)$ in actuarial literature is ${}_{T-t}p_{x+t}$.

As a canonical actuarial contract, we henceforth focus on deferred life annuities. These contracts are fundamental to valuation of defined benefit pension plans, which normally begin paying annuitants at retirement age (typically age 65) and continue until their death, possibly with survivor benefits. (For valuation purposes the payment is assumed to end at some pre-specified upper age $\bar{x}$, e.g. 100 or 110). A major problem of interest is valuing such life annuities for current plan participants who are still working, i.e. under age 65. Because this requires making longevity projections many decades into the future, longevity risk

becomes a crucial part of risk management. The net present value of a life annuity at date T is

$$\begin{aligned} a(Z(T), T, x) &\doteq \sum_{s=1}^{\infty} B(T, T+s)\mathbb{P}(\tau_x > T+s \mid \mathcal{F}_T) \\ &= \sum_{s=1}^{\bar{x}-x-T} e^{-rs} P(Z(T); T, T+s, x), \end{aligned} \tag{4.9}$$

where we emphasize that the random mortality shocks come from Z. Finally, the net present value at $t = 0$ is $NPV \doteq \mathbb{E}[e^{-rT} \cdot a(Z(T), T, x)]$, which can be seen as an instance of (4.3) that includes discounting and integrating over the density of $Z(T)$. Except for the simplest models, the survival probability $P(z; \cdot)$ is not analytically known and hence neither is (4.9) or the NPV. Without a representation for $z \mapsto a(z, T, x)$ one is then forced to resort to approximations for all the basic tasks of pricing, hedging, asset liability management, or solvency capital computation. The discussed nested simulation takes the form of first approximating $a(z^{(1)}, T, x)$ for some representative scenarios $(z^{(1)}, \ldots, z^{(n)})$, and then further manipulating the resulting "empirical" distribution of $(a(z^{(1)}, T, x), \ldots, a(z^{(n)}, T, x))$. Emulation provides a principled statistical framework for optimizing, assessing and improving such two-level simulations.

Remark. As mentioned, estimation of $a(\cdot, T, x)$ is usually a building block embedded in a larger setting which requires repeated evaluation of the former quantity. For instance, Bauer et al. (2012b) addresses nested Monte Carlo simulations in calculating the present value of life-annuity-like instruments in the calculation of solvency capital requirements.

Variable Annuities

Another broad class of insurance products where emulation is pertinent are variable and equity-indexed annuities (EIA's). EIA contracts include payments that are tied to a risky asset (such as a stock index), and in addition frequently feature deferred annuities, payment guarantees, withdrawal options and death benefits. Such riders make the cashflows of EIA's path-dependent. As a result, valuation of EIA's is well-suited for emulation because the distribution of cashflows is generally accessible only via Monte Carlo simulation. See Bauer et al. (2008) and Bacinello et al. (2011) for details and numeric methods for pricing of various EIA's.

For some EIA's our methods described above are directly transferrable. For instance, one commonly offered product is the *Guaranteed Annuity Option* (GAO), which has payoff

$$C(T) = \max\left(gV(T)a(Z(T),T,x), V(T)\right) = V(T) + gV(T)\left(a(Z(T),T,x) - 1/g\right)^{+} \tag{4.10}$$

where T is the expiration date, x is policyholder age at inception, g is a guaranteed annuity rate, and $1/g$ is the strike of the Call option on the deferred annuity value. The goal of GAO is to protect the buyer from a rise in annuitization costs at T, due to either low interest rates or dramatically improved longevity. For further details on GAOs, see Ballotta and Haberman (2006) who provide details and numeric results using Monte Carlo. In practice $V(\cdot)$ is a function of a risky asset, such as $V(T) = \max_{t \le T} S(t)$, for a mutual fund $S(\cdot)$. Equation (4.10) illustrates that the GAO NPV $\mathbb{E}[e^{-rT}C(T)|\mathcal{F}(0)]$ can be estimated through nested Monte

Carlo, where the inner simulations determine $a(Z(T), T, x)$. Consequently, our method can efficiently estimate $C(T)$ by providing an emulator for $a(Z(T), T, x)$. We revisit more complex EIA's in Section 4.7.

4.2.2 Stochastic Mortality

We concentrate on discrete-time mortality models which are easier to calibrate to the discrete mortality data, typically aggregated into annual intervals. The common assumption is that the central force of mortality remains constant through a given calendar year, so that for all $0 \leq s, u \leq 1$, we have $m(t+s, x+u) = m(t, x)$. Therefore

$$P(Z(u); t, T, x) = \mathbb{E}\left[\exp\left(-\sum_{s=t+1}^{T} m(s, x+s)\right)\middle| Z(u)\right], \quad u \leq t < T. \tag{4.11}$$

Thus, $P(Z(T); T, T+s, x+T)$ becomes a functional of the trajectory of Z between T and $T+s$.

Three major approaches to stochastic mortality have been put forward in the literature. The first approach, pioneered by Lee and Carter (1992), directly treats $m(t, x)$ as a product of individual stochastic processes, e.g. ARIMA time-series. This setup allows incorporating demographic insights, as well as disentangling age, period and cohort effects in future forecasts. To wit, the popular age-period-cohort (APC) mortality models assume that (see 4.9 for more details)

$$\log m(t, x) = \beta_x^{(1)} + \frac{1}{n_a}\kappa^{(2)}(t) + \frac{1}{n_a}\gamma^{(3)}(t-x), \tag{4.12}$$

where $\kappa^{(2)}$ and $\gamma^{(3)}$ are stochastic processes and n_a is the number of ages that x can take in fitting. In this case, the state process $Z(t)$ depends on current and potentially past values of $\kappa^{(2)}$ and $\gamma^{(3)}$. Attempts to understand the statistical validity of such models have been done by, for example, Lee and Miller (2001), Brouhns et al. (2002a), Booth et al. (2002), Czado et al. (2005a), Delwarde et al. (2007), and Li et al. (2009). Extensions of the Lee Carter model have appeared in Renshaw and Haberman (2006), Hyndman and Ullah (2007a), Plat (2009), Debonneuil (2010), and Cairns et al. (2011b).

None of these models admit closed form expressions for survival probabilities $P(z;\cdot)$. Consequently, several authors have proposed approximation methods. Coughlan et al. (2011) used a bootstrapping approach, while Cairns et al. (2014) derived an analytic approximation, commenting that industry practice is to utilize deterministic projections. Monte Carlo simulation has been applied in Bauer et al. (2012b) among others.

The second approach, due to Cairns et al. (2006) (CBD), generates a stochastic model for the survival probability (4.11), allowing for straightforward pricing of longevity-linked products; however, it is more difficult to calibrate and to obtain reasonable forecasts for future mortality experience in a population as a whole. The third approach works with forward mortality rates Bauer et al. (2012a), borrowing ideas from fixed income markets. Forward models give a holistic view of how the mortality curves can evolve over time, and presents a dynamically consistent structure for mortality forecasting. Once again however, such models do not provide closed-form expressions for (4.11) and hence require further manipulation for pricing purposes.

4.2.3 Bias/Variance Trade-Off

With a view towards approximating (4.9), it is imperative to first quantify the resulting quality of an approximation. The standard statistical approach is to use the framework of mean squared error. Fix z and let $a(z) \equiv a(z, T, x)$ be the true value of a life annuity conditional on state $Z(T) = z$. If $a(z)$ is being estimated by $\hat{a}(z)$, then

$$\text{IMSE}(\hat{a}) \doteq \mathbb{E}\left[(\hat{a} - a)^2\right], \qquad \text{Bias}(\hat{a}) = \mathbb{E}\left[\hat{a} - a\right], \tag{4.13}$$

where the averaging is over the sampling distribution (i.e. different realizations of data used in constructing it) of $\hat{a}(z)$.

Starting with (4.13) leads to the fundamental bias/variance trade-off. At one end of the spectrum, a Monte Carlo estimate as in (4.6) has zero bias but carries a high variance. At the opposite end, an analytic approximation has zero variance, but will have a non-zero bias that cannot be alleviated (whereas the Monte Carlo IMSE will go to zero as the size of the dataset grows $N_{tr} \to \infty$) even asymptotically. Because low variance is often preferred practically, analytic methods have remained popular. Cairns et al. (2014) echoes that it is usual practice in industry to use a deterministic projection of mortality rates rather than use a simulation approach. The basic idea for the deterministic approximations is that if $\widehat{m}(t, x)$

is an unbiased estimate for $m(t,x)$, then

$$\begin{aligned} P(Z(u);t,T,x) &= \mathbb{E}\left[\exp\left(-\sum_{s=t+1}^{T} m(s,x+s)\right)\middle| Z(u)\right] \\ &\approx \exp\left(-\sum_{s=t+1}^{T}\mathbb{E}\left[m(s,x+s)\mid Z(u)\right]\right) \\ &= \exp\left(-\sum_{s=t+1}^{T}\widehat{m}(s,x+s;Z(u))\right). \end{aligned} \tag{4.14}$$

Using the estimate for $P(Z(u);\cdot)$ in (4.14) one can then approximate $a(Z(T),T,x)$ term-wise. Jensen's inequality implies that $\exp\left(-\sum_{s=t+1}^{T}\widehat{m}(s,x+s)\right) > P(Z(u);t,T,x)$. Consequently, any such approximation is guaranteed to be biased high for the survival probabilities (and subsequently the annuity values).

Analytic approximations can be very powerful and of course very fast but they carry two major disadvantages. One is the need to *derive* a suitable estimator $\widehat{m}$. This may be possible in a simple model (e.g. low-dimensional Z with linear dynamics, like in the original Lee-Carter model), but otherwise may require a lot of off-line labor, leading to unnecessary focus on simplifications at the expense of calibration and risk management consistency. Second, the degree of accuracy of the approximation is unknown. Indeed, there is generally not much that is available about empirical accuracy of the right-hand-side in (4.14) for a given model, leaving the user in the dark about how much error is being made. This issue is very dangerous, since potentially major mis-valuations may creep up unbeknownst to the risk manager.

To remedy the above shortcomings, while still maintaining significant variance reduction compared to plain MC, we advocate the use of statistical emula-

tors. Emulation is a data-driven approximation technique that originated in the machine learning and simulation literatures. Emulators allow posterior quantification of accuracy (via standard error or Bayesian posterior variance), and their implementation does not require any simplifications of the mortality model. An additional advantage is that one can directly approximate $z \mapsto a(z, T, x)$ without having to do intermediate approximations of the survival probabilities (which inevitably lead to further error compounding). As we demonstrate in the case studies, statistical models for $a(z)$ can indeed efficiently address the bias/variance trade-off, by maintaining negligible bias and small variance, leading to improved IMSE metrics.

4.3 Statistical Emulation

The idea of emulation is to replace the computationally expensive process of running a Monte Carlo sub-routine to evaluate $f(z)$ for each new site z with a cheap-to-evaluate surrogate model that statistically predicts $f(z)$ for any $z \in \mathbb{R}^d$ based on results from a training dataset. At the heart of emulation is statistical learning. Namely, the above predictions are based on first obtaining pathwise estimates $y^{(n)} = F(T, z^{(n)})$, $n = 1, \ldots, N_{tr}$ for a set of training locations, called a design $\mathcal{D} \doteq (z^{(1)}, \ldots, z^{(N_{tr})})$. Next, one regresses $\{y^{(n)}\}$ against $\{z^{(n)}\}$ to "learn" the response surface $\hat{f}(\cdot)$. The regression aspect allows to borrow information across different scenarios starting at various sites. This reduces computational budget compared to the nested simulation step of independently making N_{tr} pointwise estimates $f(z^{(n)})$ by running $N_{tr,2}$ scenarios from *each* site $z^{(n)}$. The conceptual

need for regression is two-fold. First, the emulator is used for interpolation, i.e. using existing design to make predictions at new sites z. In contrast, plain Monte Carlo only predicts at $z^{(n)}$'s. Second, like in the classical approach, the emulator *smoothes* the Monte Carlo noise from sampling trajectories of $\{Z(s), s > T\}$.

Formally, the statistical problem of emulation deals with a sampler (or oracle)

$$Y(z) = f(z) + \epsilon(z), \tag{4.15}$$

where we identify $f(z) \equiv a(z, T, x)$ with the unknown *response surface* and ϵ is the sampling noise, assumed to be independent and identically distributed across different calls to the oracle. We make the assumption $\epsilon(z) \sim N(0, \tau^2(z))$, where $\tau^2(z)$ is the sampling variance that depends on the location z. Emulation now involves the (i) experimental design step of proposing a design $\mathcal{D}$ that forms the training dataset, and (ii) a learning procedure that uses the queried results $(z^{(n)}, y^{(n)})_{n=1}^{N_{tr}}$, with the $y^{(n)}$ being realizations of (4.15) given $z^{(n)}$, to construct a fitted response surface $\hat{f}(\cdot)$. The fitting is done by specifying the approximation function class $\hat{f} \in \mathcal{H}$, and a loss function $L(\hat{f}, f)$ which is to be minimized. The loss function measures the relative accuracy of $\hat{f}$ vis-a-vis the ground truth; in this paper we focus on the mean-squared approximation error

$$L(\hat{f}, f) \doteq \int_{\mathbb{R}^d} |\hat{f}(z) - f(z)|^2 dz. \tag{4.16}$$

Because the true f is unknown, the definition of $L(\hat{f}, f)$ cannot be operationalized and instead a proxy based on the uncertainty (such as Bayesian posterior uncertainty or standard error) surrounding $\hat{f}$ is applied. Also, since the structure

of f is unknown, it is desirable that the approximation class $\mathcal{H}$ is dense, i.e. has a sufficiently rich architecture to approximate any f to an arbitrary degree of accuracy. To this end, we concentrate on kernel regression methods, namely linear smoothers. In the next subsections we introduce two such regression families, smoothing splines and kriging (Gaussian process) models.

Remark. In this paper we focus on the original task of producing an accurate approximation to f everywhere. In some contexts, accuracy is judged not globally, but locally, so that a differentiated accuracy measure is used. For example, in VaR applications, the model for f must be accurate in the left-tail, but can be rather rough in the right-tail. In this case, (4.16) can be replaced by a weighted loss metric, see e.g. Liu and Staum (2010).

4.3.1 Emulators based on Spline Models

We generate emulators $\hat{f}(\cdot)$ using a regularized regression criterion. To wit, given a smoothing parameter $\lambda \geq 0$ we look for the minimizer $\hat{f} \in \mathcal{H}$ of the following penalized residual sum of squares problem

$$RSS(f, \lambda) = \sum_{n=1}^{N_{tr}} \{y^{(n)} - f(z^{(n)})\}^2 + \lambda J(f), \tag{4.17}$$

where $J(f)$ is a penalty or regularization function. We concentrate on the case where the approximation class has a reproducing kernel Hilbert space (RKHS) structure which also generates $J(f)$. Namely, there exists an underlying positive definite kernel $C(z, z')$ such that $\mathcal{H}_C = \text{span}(C(\cdot, z) : z \in \mathbb{R}^d)$ is the Hilbert space generated by C and $J(f) = \|f\|^2_{\mathcal{H}_C}$. The representer theorem implies that the

minimizer of (4.17) has an expansion in terms of the eigen-functions

$$\hat{f}(z) = \sum_{j=1}^{N_{tr}} \alpha_j C(z, z^{(j)}), \tag{4.18}$$

relating the prediction at z to the kernel function sampled at the design sites $z^{(j)}$'s.

Our first family are smoothing (or thin-plate) splines that take

$$J(f) = \int_{\mathbb{R}^d} \left[\sum_{i,j=1}^{d} \frac{\partial}{\partial z_i} \frac{\partial}{\partial z_j} f(z) \right] dz, \tag{4.19}$$

and $\mathcal{H}$ as the set of all twice continuously-differentiable functions. It is known (Hastie et al., 2009, Chapter 5) that in this case the underlying kernel is given by $C(z, z') = \|z - z'\|^2 \log \|z - z'\|$, where $\|\cdot\|$ denotes the Euclidean norm in $\mathbb{R}^d$. The resulting optimization of (4.17) along with (4.19) gives a smooth response surface which is called a thin-plate spline (TPS), and has the explicit form

$$f(z) = \beta_0 + \boldsymbol{\beta}^T \vec{z} + \sum_{j=1}^{N_{tr}} \alpha_j \|z - z^{(j)}\|^2 \log \|z - z^{(j)}\|, \tag{4.20}$$

with $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_d)^T$.

In 1-d, the penalized optimization reduces to

$$\inf_{f \in \mathcal{C}^2} \sum_{i=1}^{N_{tr}} \{y^{(n)} - f(z^{(n)})\}^2 + \lambda \int_{\mathbb{R}} \{f''(u)\}^2 du. \tag{4.21}$$

The summation in (4.21) is a measure of closeness of data, while the integral penalizes the fluctuations of f. Note that $\lambda = \infty$ reduces to the traditional least squares linear fit $\hat{f}(z) = \beta_0 + \beta_1 z$ since it introduces the constraint $f''(z) = 0$.

The resulting solution is an expansion in terms of natural cubic splines, i.e. the minimizer $\hat{f}$ of (4.21) is a piecewise cubic polynomial that has continuous first and second derivatives at the design sites $z^{(n)}$, and is linear outside of the design region.

Several methods are available to choose the smoothing parameter λ, including cross-validation or MLE (Hastie et al., 2009, Chapter 5). A common parametrization is through the effective degrees of freedom statistic df_λ. We use the `R` package "fields" Nychka et al. (2015) to fit multi-dimensional thin plate splines, and the base `smooth.spline` function for the one-dimensional case.

4.3.2 Kriging Surrogates

A kriging surrogate assumes that f in (4.15) has the form

$$f(z) = \mu(z) + X(z), \tag{4.22}$$

where $\mu : \mathbb{R}^d \to \mathbb{R}$ is a trend function, and X is a mean-zero square-integrable process. Specifically, X is assumed to be a realization of a Gaussian process with covariance kernel C. The role of C is identical to the regularized regression above, i.e. C generates the approximating class $\mathcal{H}_C$ that X is assumed to belong to.

However, kriging also brings a Bayesian perspective, treating X as a random function to be learned, and estimation as computing the posterior distribution of X given the collected data $\mathbf{y} \doteq (y^{(1)}, \ldots, y^{(N_{tr})})$. The RKHS framework implies that the posterior mean (more precisely its maximum a posteriori estimate) of $X(z)$ coincides with the regularized regression prediction from the previous

section. In the Bayesian framework, C is interpreted as the covariance kernel, $C(z, z') = Cov(f(z), f(z'))$ as $f(\cdot)$ ranges over $\mathcal{H}_C$. Assuming that the noise $\epsilon(z)$ is also Gaussian implies that $X(z)|\mathbf{y} \sim N(m(z), s^2(z))$ has a Gaussian posterior, which reduces to computing the kriging mean $m(z)$ and kriging variance $s^2(z)$.

In turn, the kriging variance $s^2(z)$ offers a principled empirical estimate of model accuracy, quantifying the approximation quality. In particular, one can use $s^2(z)$ as the proxy for the MSE of $\hat{f}$ at z. Integrating $s^2(z^{(n)})$ over the design locations then yields an assessment regarding the error of (4.3).

Simple Kriging

Simple kriging (SK) assumes that the trend $\mu(z)$ is known. By considering the process $f(z) - \mu(z)$, we may assume without loss of generality that $f(z)$ is centered at zero and $\mu \equiv 0$. The resulting posterior mean and variance are then Roustant et al. (2012a)

$$\begin{cases} m_{SK}(z) \doteq \mathbf{c}(z)^T \mathbf{C}^{-1} \mathbf{y}; \\ s^2_{SK}(z) \doteq C(z, z) - \mathbf{c}(z)^T \mathbf{C}^{-1} \mathbf{c}(z), \end{cases} \tag{4.23}$$

where $\mathbf{c}(z) = \left(C(z, z^{(n)})\right)_{1 \le n \le N_{tr}}$ and

$$\mathbf{C} \doteq \left[C(z^{(i)}, z^{(j)})\right]_{1 \le i,j \le N_{tr}} + \mathbf{\Delta}, \tag{4.24}$$

with $\mathbf{\Delta}$ the diagonal matrix with entries $\tau^2(z^{(1)}), \ldots, \tau^2(z^{(N_{tr})})$.

Universal Kriging

Universal kriging (UK) generalizes (4.22) to the case of a parametric trend function of the form $\mu(z) = \beta_0 + \sum_{j=1}^{p} \beta_j h_j(z)$ where β_j are constants to be estimated, and $h_j(\cdot)$ are given basis functions. The coefficient vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is estimated simultaneously with the Gaussian process component $X(z)$. A common choice is first-order UK that uses $h_j(z) = z_j$ for $j = 1, \ldots, d$. Another common choice is zero-order UK, also known as Ordinary Kriging (OK) that takes $\mu(z) = \beta_0$ a constant to be estimated.

If we let $\mathbf{h}(z) \doteq (h_1(z), \ldots, h_p(z))$ and $\mathbf{H} \doteq \left(\mathbf{h}(z^{(1)}), \ldots, \mathbf{h}(z^{(N)})\right)$, then the universal kriging mean and variance at location z are Roustant et al. (2012a)

$$m_{UK}(z) = \mathbf{h}(z)^T \hat{\boldsymbol{\beta}} + \mathbf{c}(z)^T \mathbf{C}^{-1}(\mathbf{y} - \mathbf{H}\hat{\boldsymbol{\beta}}); \tag{4.25}$$

$$s^2_{UK}(z) = s^2_{SK}(z) + \left(\mathbf{h}(z)^T - \mathbf{c}(z)^T \mathbf{C}^{-1}\mathbf{H}\right)^T \left(\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H}\right)^{-1} \left(\mathbf{h}(z)^T - \mathbf{c}(z)^T \mathbf{C}^{-1}\mathbf{H}\right),$$

where the best linear estimator of the trend coefficients $\boldsymbol{\beta}$ is given by the usual linear regression formula $\hat{\boldsymbol{\beta}} \doteq \left(\mathbf{H}^T \mathbf{C}^{-1} \mathbf{H}\right)^{-1} \mathbf{H}^T \mathbf{C}^{-1} \mathbf{y}$.

The combination of trend and Gaussian process (GP) model offers an attractive framework for fitting a response surface. The trend component allows to incorporate domain knowledge about the response, while the GP component offers a flexible nonparametric correction. One strategy is to specify a known trend (coming from some analytic approximation) and fit a GP to the residuals, yielding a Simple Kriging setup. Another strategy is to take a low-dimensional parametric approximation, such as a linear function of Z-components, and again fit a GP to the residuals, leading to a Universal Kriging setup.

Covariance kernels and parameter estimation

The covariance function $C(\cdot, \cdot)$ is a crucial part of a Kriging model. In practice, one usually considers spatially stationary or isotropic kernels,

$$C(z, z') \equiv c(z - z') = \sigma^2 \prod_{j=1}^{d} g((z - z')_j; \theta_j),$$

reducing to the one-dimensional base kernel g. Below we use the power exponential kernels $g(h; \theta) = \exp\left(-\left(\frac{|h|}{\theta}\right)^p\right)$. The hyper-parameters θ_j are called characteristic length-scales and can be informally viewed as roughly the distance you move in the input space before the response function can change significantly (Rasmussen and Williams, 2006, Ch 2). The user-specified power $p \in [1, 2]$ is usually taken to be either $p = 1$ (the exponential kernel) or $p = 2$ (the Gaussian kernel). Fitting a kriging model requires picking a kernel family and the hyper-parameters σ_j, θ_j. Two common estimation methods are maximum likelihood, using the likelihood function based on the distributions described above, and penalized MLE. Either case leads to a nonlinear optimization problem to fit θ_j and process variance σ^2. One can also consider Bayesian Kriging, where trend and/or covariance parameters have a prior distribution, see Helbert et al. (2009). We utilize the `R` package "DiceKriging" Roustant et al. (2012a) that allows fitting of SK and UK models with five options for a covariance kernel family, and several options on how the hyper-parameters are to be estimated.

Batching

To construct an accurate emulator for $f(\cdot)$ it is important to have a good estimate of the sampling noise $\tau^2(z)$. Typically this information is not available to the modeler a priori. One of the advantages of plain nested Monte Carlo is that generating N_{in} scenarios from a fixed $z^{(n)}$ gives natural empirical estimates *both* for $f(z^{(n)})$ and $\tau^2(z^{(n)})$. To mimic this feature, we therefore consider batched or replicated designs $\mathcal{D}$. To wit, given a total budget of $N_{tr} = N_{tr,1} \cdot N_{tr,2}$ training samples, we allocate them into $N_{tr,1}$ distinct design sites $z^{(1)}, \ldots, z^{(N_{tr,1})}$, and then generate $N_{tr,2}$ trajectories from each $z^{(n)}$. Next, the above batches are aggregated into

$$y^{(n)} \doteq \frac{1}{N_{tr,2}} \sum_{j=1}^{N_{tr,2}} F(T, z^{(n),j}(\cdot)); \tag{4.26}$$

$$\hat{\tau}^2(z^{(n)}) \doteq \frac{1}{N_{tr,2}-1} \sum_{j=1}^{N_{tr,2}} \left\{ y^{(n)} - F(T, z^{(n),j}(\cdot)) \right\}^2, \tag{4.27}$$

and the resulting dataset $\{z^{(n)}, y^{(n)}, \hat{\tau}^2(z^{(n)})\}$, $n = 1, \ldots, N_{tr,1}$ is used to fit a kriging model for $\hat{f}$, with $\hat{\tau}^2(z^{(n)})/N_{tr,2}$ proxying the simulation variance at $z^{(n)}$.

The efficient allocation between $N_{tr,1}$ and $N_{tr,2}$ was analyzed in Broadie et al. (2011) for a related risk management problem and it was shown that the optimal choices satisfy

$$N_{tr,1} \propto N_{tr}^{2/3}, \qquad N_{tr,2} \propto N_{tr}^{1/3}. \tag{4.28}$$

This is also the allocation we pursue in this paper, so that there are relatively many more design sites than replications in each batch.

4.3.3 Least Squares Monte Carlo

Another example of an emulator is the so-called Least Squares Monte Carlo (LSMC) approach introduced to the actuarial context in Bacinello et al. (2010, 2011). LSMC is similar to the kriging and spline emulation, except now the surrogate is based on a linear model given through a prespecified set of basis functions $\mathbf{h} = (h_1, \ldots, h_p)$. Thus, the predicted $Y(z)$ is

$$\hat{y}(z) \doteq \hat{\boldsymbol{\beta}}\mathbf{h}(z), \tag{4.29}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ are the regression coefficients obtained through ordinary least squares regression of the sampled $\{y^{(n)}\}$ against the design $\{z^{(n)}\}$:

$$\hat{\boldsymbol{\beta}} \doteq \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^p} \sum_{j=1}^{N_{tr,1}} \left(y^{(j)} - \boldsymbol{\beta}\mathbf{h}(z^{(j)})\right)^2. \tag{4.30}$$

See Bacinello et al. (2010) for further details and literature on this topic with applications to pricing life insurance products with a surrender option, and Bauer et al. (2012b) for an application to Solvency II. In Section 4.5 we apply this technique as an additional comparison to kriging and spline emulation.

A major challenge with LSMC is that the modeler must a priori specify the number of basis functions p and their functional form. This is difficult to do in multi-dimensional settings. In particular, using too few basis functions risks introducing significant bias in $\hat{y}^{LSMC}$, while too many basis functions will lead to over-fitting. Statistically, the quality of the LSMC estimator depends on the distance between the true response map Y and the manifold $span(h_1(z), \ldots, h_p(z))$

generated by the basis functions. A simple strategy is to use polynomials up a given degree $\bar{d}$, e.g a quadratic model with $\mathbf{h} = (z_1, z_1^2, z_2, z_2^2, z_1 z_2, \ldots)$ which is the approach we pursue in Section 4.5.

4.3.4 Experimental Design

Several approaches are possible for constructing the training design $\mathcal{D}$. First, one may generate an empirical design by independently sampling $z^{(n)} \sim Z(T)|Z(0)$. This allows to emulate the conditional density $p_T(z|Z(0))$ which is advantageous for computing an expectation like in (4.3). Second, one may generate a random $\mathcal{D}$ using some other proposal density $z^{(n)} \sim Q$. For example, a uniform proposal density (i.e. $z^{(n)}$ i.i.d. uniform in some domain $D \subseteq \mathbb{R}^d$) yields a basic space filling experimental design of arbitrary size. A more structured (but still random) design can be obtained via Latin Hypercube Sampling (LHS) techniques Wyss and Jorgensen (1998). Roughly speaking, LHS builds a regular d-dimensional lattice and then attempts to equidistribute $N_{tr,1}$ sites among the resulting hypercubes. Within each selected hypercube the design site is placed uniformly.

Third, one can use a deterministic design, such as a latticed grid, or a quasi-Monte Carlo (QMC) sequence. Deterministic designs ensure a space-filling property and easy reproducibility. For example, the Sobol sequence Sobol (1998) redistributes a uniform binary grid to produce a grid that is maximally equidistributed. Compared to LHS, use of QMC is faster (as it can be directly hard-coded) and can be manually tweaked as needed. Both methods reduce Monte Carlo variance of $\hat{f}$ relative to empirical $\mathcal{D}$. Lastly we mention that the typical domain of $Z(T)$ is in principle unbounded, e.g. $\mathbb{R}^d_+$. This is not an issue for empirical design construc-

tion; for LHS and QMC methods, one must restrict to an appropriate bounding domain $D \in \mathbb{R}^d$ before generating $\mathcal{D}$.

Remark. Depending on the context, the design $\mathcal{D}$ might need to be spatially non-uniform. For example, if using a deterministic design for computing (4.3), it may be preferable to capture the correlation structure among the components of $Z(T)$, or to up-weigh the regions most likely for $Z(T)$. If one is estimating a quantile or tail expectation, $\mathcal{D}$ should preferentially cover the extreme values of the distribution of $Z(T)$; in that situation, an empirical design would be inappropriate. See for instance Liu and Staum (2010) who investigated evaluation of expected shortfall of stock portfolios using two-stage experimental design. To wit, starting with an initial space-filling design Liu and Staum (2010) first augmented with more tail scenarios and then further allocated inner simulation budget based on a combination of surrogate variance and tail weight.

Generating Longevity Scenarios

Construction of an emulator entails the basic building block of generating a longevity scenario $\{Z(t), t = 0, \ldots\}$. In the simplest setting, this just requires to generate and manipulate a sequence of i.i.d Uniform draws that describe the random increments (of the components) of Z. However, typically the stochastic model used also includes parameters that must be estimated or calibrated. This aspect becomes nontrivial when future longevity projections are made, whereby model re-fitting may be carried out. Re-fitting introduces path-dependency, making parameters dynamic quantities that might need to be included in Z. For example, Cairns et al. (2014) advocate the PPC (partial parameter certain) sce-

nario generation that breaks the overall simulation into two pieces of $[0, T]$ and $[T, \infty)$. With PPC, one initially calibrates the model at $t = 0$ using past mortality data and then simulates up to time T. The simulated scenario is then appended to the historical data, so that the simulation becomes the new "history" from time 0 to time T. The model parameters are then re-fitted at T and the resulting, modified longevity dynamics of Z are used to simulate beyond T. The idea of PPC is to capture some memory of mortality evolution, in essence removing some of the presumed Markovian structure. Under PPC the refitted parameters are blended into $Z(T)$ since they affect the resulting $F(T, Z(\cdot))$.

Conversely, in the interest of dimension reduction, one could drop some components of the full state space when constructing the emulator. To do so, one may analyze what dynamic variables materially impact annuity values, for example via some simple regression models to test for statistical significance.

4.3.5 Fitting and Evaluation of Emulators

To fit an emulator for a given simulation budget N_{tr}, we first decompose $N_{tr} = N_{tr,1} \times N_{tr,2}$ and then construct an experimental design $\mathcal{D}$ of size $N_{tr,1}$ using one of the methods in Section 4.3.4. Each site in $\mathcal{D}$ then spawns $N_{tr,2}$ trajectories that are batched together as in (4.26).

Fitting is done in `R` using the mentioned publicly available packages. For kriging we use the default setting of the `km` function in the DiceKriging package.

Given $\hat{a}(z, T, x)$ we evaluate its performance across a test set

$$\mathcal{D}^{test} = (z^{(1)}, \ldots, z^{(N_{out})})$$

of N_{out} locations. Note that $\mathcal{D}^{test}$ is distinct from the training set $\mathcal{D}$. In line with (4.3) we use an empirical testing set $\mathcal{D}^{test}$: $z^{(n)} \sim Z(T)|Z(0)$. Since the true values $a(z,T,x)$ are not available, we benchmark against an (expensive) gold standard estimate $\hat{a}^{MC}(z,T,x)$ that is described below. In particular, we record the integrated MSE and Bias statistics from (4.13), namely

$$\widehat{\text{IMSE}}\left(\hat{a}\right) = \frac{1}{N_{out}} \sum_{n=1}^{N_{out}} \left(\hat{a}(z^{(n)},T,x) - \hat{a}^{MC}(z^{(n)},T,x)\right)^2; \tag{4.31}$$

$$\widehat{\text{Bias}}\left(\hat{a}\right) = \frac{1}{N_{out}} \sum_{n=1}^{N_{out}} \left[\hat{a}(z^{(n)},T,x) - \hat{a}^{MC}(z^{(n)},T,x)\right]. \tag{4.32}$$

For benchmarking, we use a high-fidelity nested Monte Carlo approach (4.5)-(4.6). While expensive, it is a simple, asymptotically consistent, unbiased estimator. Specifically, for valuing annuities, $\hat{a}^{MC}(z^{(n)},T,x)$ is obtained by averaging $N_{in} = 10^5$ scenarios of $\{Z(s), s > T\}$ at each $z^{(n)} \in \mathcal{D}^{test}$. Unless indicated otherwise, we use $N_{out} = 1000$, so that the overall budget of $\hat{a}^{MC}$ is $\mathcal{O}(N_{in} \times N_{out})$. We then compare against emulators that use $N_{tr} \in [100, 8000]$, which yields an efficiency gain on the order of 10-50 times speed-up. We also compare against deterministic estimators that require no training at all (but do need an analytic derivation), and take just $\mathcal{O}(N_{out})$ budget to make predictions for the outer N_{out} simulations to evaluate (4.31).

4.4 Case Study: Predicting Annuity Values under a Lee-Carter with Shocks Framework

Our first case study features a relatively simple one-dimensional state Z that allows to visualize the emulator structure and its experimental design. As we shall see, for such more straightforward settings, most approximation methods work well, so our emphasis is on further explaining the kriging emulators rather than maximizing performance.

Chen and Cox (2009) introduced a mortality model based on the traditional Lee Carter set-up:

$$\log m(t,x) = \beta^{(1)}(x) + \beta^{(2)}(x)\kappa^{(2)}(t). \tag{4.33}$$

This is the same as the APC model (M2) in 4.9 without the cohort term. In the Chen-Cox model, $\beta^{(1)}(x)$ and $\beta^{(2)}(x)$ are deterministic vectors capturing age effects, and $\kappa^{(2)}(t)$ is a stochastic process capturing the period effect with dynamics

$$\kappa^{(2)}(t+1) = \kappa^{(2)}(t) + \mu^{(1)} + \xi^{(1)}(t+1) + [\xi^{(2)}(t+1) - \xi^{(2)}(t)], \tag{4.34}$$

where $\xi^{(1)}(t) \sim N(0, \sigma^{(1)})$ and $\xi^{(2)}(t)$ has an independent zero-modified normal distribution with $\mathbb{P}(\xi^{(2)}(t) = 0) = 1-p$, and Gaussian parameters $(\mu^{(2)}, \sigma^{(2)})$. The motivation for (4.34) is to incorporate idiosyncratic mortality shocks represented by $\xi^{(2)}$, that occur with probability p any given year and have a random magnitude with distribution $N(\mu^{(2)}, \sigma^{(2),2})$. Such shocks, representing natural or geopolitical catastrophes, are temporary and last just a single period, hence subtraction of the

last term $-\xi^{(2)}(t)$ in (4.34). Due to this term, it would appear that the model has a two-dimensional state space $\{\kappa^{(2)}(t), \xi^{(2)}(t)\}$. However, we note that it is sufficient to generate scenarios starting with $\kappa^{(2)}(T)$ and assuming $\xi^{(2)}(T) = 0$ (no shock in year T). Then after estimating $f(\kappa) = \mathbb{E}[F(T, \kappa^{(2)}(\cdot))|\kappa^{(2)}(T) = \kappa, \xi^{(2)}(T) = 0]$, one easily obtains in case of year-T shocks $\mathbb{E}[F(T, \kappa^{(2)}(\cdot)|\kappa^{(2)}(T) = \kappa, \xi^{(2)}(T) = \xi] = f(\kappa - \xi)$, reducing to the prediction of "unshocked" values.

The presence of idiosyncratic shocks in $m(t, x)$ renders the corresponding survival probability analytically intractable. However, the linear dynamics of $\kappa^{(2)}$ in (4.34) allows to obtain the following deterministic estimator for future mortality rates.

Lemma 1. *Let $Z(s) = \{\kappa^{(2)}(s), \xi^{(2)}(s)\}$. Under the Chen-Cox model, the following holds:*

$$\mathbb{E}[\kappa^{(2)}(t) \mid Z(s)] = \kappa^{(2)}(s) + (t-s)\mu^{(1)} + \mu^{(2)}p - \xi^{(2)}(s), \qquad 0 \leq s \leq t < \infty. \quad (4.35)$$

The proof can be found in 4.10. Substituting (4.35) into (4.33) yields the following estimator for $\mathbb{E}[m(T+s, x) \mid \kappa^{(2)}(T), \xi^{(2)}(T)]$:

$$\widehat{m}(T+s, x) \doteq \exp\left(\beta^{(1)}(x) + \beta^{(2)}(x)\left(\kappa^{(2)}(T) + s\mu^{(1)} + \mu^{(2)}p - \xi^{(2)}(T)\right)\right). \quad (4.36)$$

4.4.1 Results

We follow Chen and Cox (2009) in using US mortality data obtained from the National Center for Health Statistics (NCHS)[1]. This dataset contains yearly age

[1]Source: `http://www.cdc.gov/nchs/nvss/mortality_tables.htm`

specific death rates for overall US population over 1900–2003. Fitting yields the random-walk parameters $\mu^{(1)} = -0.2173, \sigma^{(1)} = 0.3733$ in (4.34), as well as the estimated probability of shock as $p = 0.0436$, with jump distribution $(\mu^{(2)}, \sigma^{(2)}) = (0.8393, 1.4316)$. As expected, $\mu^{(2)} \gg 0$ is large and positive, so shocks correspond to large temporary increases in mortality. The goal is to analyze and compare the ability of kriging models and analytic estimates to predict $T = 10$-year deferred annuity values for unisex $x = 65$ year olds. Payments are cut-off at age $\bar{x} = 94$. We use a discount rate of $r = 4\%$.

	$N_{tr} = 125$		$N_{tr} = 512$		$N_{tr} = 1000$	
Type	Bias	$\sqrt{\text{IMSE}}$	Bias	$\sqrt{\text{IMSE}}$	Bias	$\sqrt{\text{IMSE}}$
Analytic	1.668e-03	2.148e-03	1.668e-03	2.148e-03	1.668e-03	2.148e-03
OK	5.145e-03	5.923e-03	1.582e-04	1.975e-03	-1.999e-04	1.634e-03
UK	5.832e-03	6.059e-03	4.816e-04	1.045e-03	-1.243e-05	7.428e-04

Table 4.1: Comparing estimators for life annuity value under the Chen-Cox model for different size of experimental design. The design $\mathcal{D}$ is constructed with $N_{tr} = N_{tr,1}^{2/3} \cdot N_{tr,2}^{1/3}$. The reported values are evaluated from a Monte Carlo benchmark, using (4.31) and (4.32). Analytic estimate is based on (4.36); universal kriging model uses first-order linear basis functions.

We fit emulators with budgets $N_{tr} \in \{125, 512, 1000\}$. The respective training designs $\mathcal{D}$ are deterministic and uniformly spaced across an appropriately chosen interval $D = [\underline{\kappa}, \bar{\kappa}]$; a fixed design minimizes sampling variation in fitting $\hat{f}(\cdot)$. Because $Z \equiv \kappa^{(2)}$ is just one-dimensional, a relatively small training budget is used. For the emulators, we fit both an ordinary kriging (OK) model with constant trend $\mu(\kappa) = \beta_0$, and first-order linear universal kriging (UK) model with $\mu(\kappa) = \beta_0 + \beta_1\kappa$. For evaluation, we fix a testing set containing $N_{out} = 50$ values of $Z(T)$, benchmarked with a nested Monte Carlo approach with $N_{in} = 10^5$ inner

simulations. Due to the very small MSE's involved, a very high-fidelity benchmark was needed (in order to isolate the MSE of the emulator from the MSE of the benchmark), leading to a very large N_{in}. To be computationally feasible, we picked a small testing set. To make sure that $\mathcal{D}^{test}$ accurately represented the distribution of $\kappa(T)$ its points were picked as the empirical $1\%, 3\%, \ldots, 99\%$ percentiles of a large sample of $\kappa(T)$. The mortality shocks associated with these percentiles were used in the comparison process.

Table 4.1 and Figure 4.1 summarize the results. We observe that there is quite a wide spread in potential future annuity prices, with differences of more than 10% (or \$1 in annuity NPV) depending on realized $Z(T)$. This confirms the significant level of longevity risk. As shown in the Figure 4.1, there is a nearly linear relationship for $z \mapsto a(z, T, x)$, which is perhaps surprising given the above range of forecasts. This strong linear trend in the response partly explains the advantage of the UK model over OK. The Figure also reflects the effect of training set size and distribution: the $N_{tr} = 512$ model performs significantly better than its $N_{tr} = 125$ counterpart. We see that all methods perform well, with IMSE's on the order of 10^{-3}. The Monte Carlo benchmark was

$$\hat{a}^{MC} = \frac{1}{N_{out}} \sum_{n=1}^{N_{out}} \hat{a}^{MC}(z^{(n)}, T, x) = \$11.91338,$$

so that the relative bias was around 0.001%, and percentage root-IMSE around 0.01%. Even though the computed biases are rather small, we remark that since pension portfolios have very large face values, the corresponding approximation errors could be financially meaningful. For example, for a modest pension fund

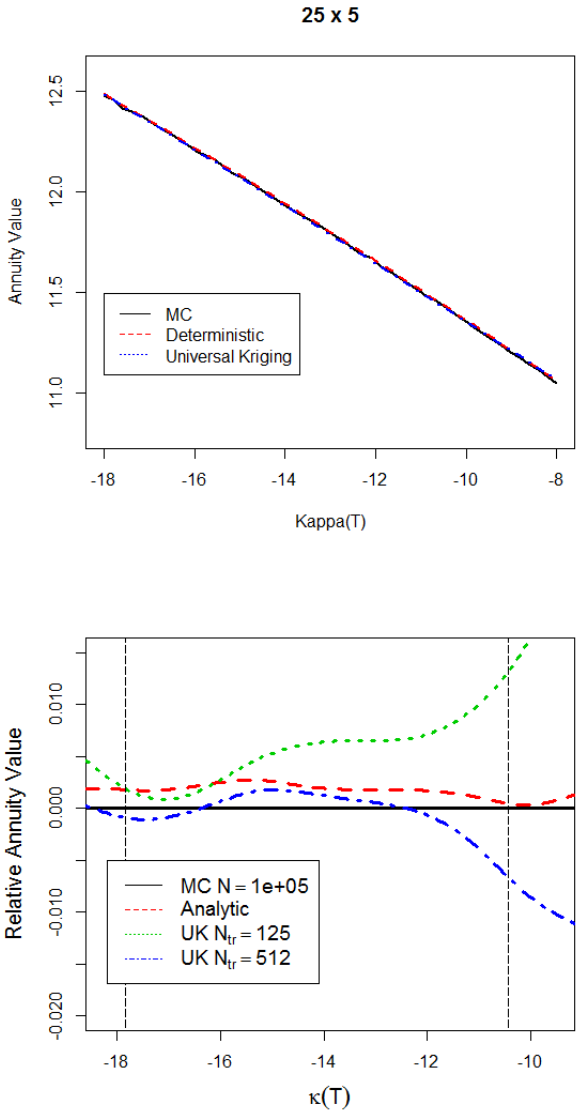

Figure 4.1: Annuity emulators in the Chen-Cox model. Left: three estimators (MC, UK w/$N_{tr} = 125$ and Analytic) of annuity value $a(\kappa^{(2)}(T))$ vs. $\kappa^{(2)}(T)$. The training design (indicated by the vertical dashed lines) is $\mathcal{D} = \{\kappa^{(2)}(T) \in (-17.5, -10)\}$ with $N_{tr,1} = 25, N_{tr,2} = 5$. Right: relative annuity values vis-a-vis the Monte Carlo benchmark $\hat{a}^{MC} = 11.91$ obtained with $N_{in} = 10^5$.

with an obligation of \$100mm, a bias of 10^{-3} implies inaccuracy of \$100k.

The right panel of Figure 4.1 provides a zoomed-in visualization of the estimators' bias relative to $\hat{a}^{MC}$. As expected, the analytic estimator based on Lemma 1 overestimates the true annuity value for all $\kappa^{(2)}(T)$. For $N_{tr} = 125$, the kriging emulator clearly has a larger MSE, and in this case it also typically overestimates $a(\kappa^{(2)}(T))$. For $N_{tr} = 512$ we observe the statistical learning taking place, as the kriging model now has an excellent fit in the middle of the figure and essentially zero bias averaging over potential values of $\kappa^{(2)}(T)$. The effect of larger training budget is confirmed in Table 4.1, with IMSE's all decreasing towards zero as N_{tr} increases.

The above analysis demonstrates that in some settings, the shape $z \mapsto a(z, T, x)$ is sufficiently simple that little modeling is required, and analytic estimators perform well (as do statistical emulators). However, we stress that there is no easy way to tell a priori that the analytic estimator would be adequate, and in any case a sufficiently large training set size will guarantee a better predictive power for the kriging models.

The one dimensional case also provides a visual representation of the effect of grid design, illustrated in Figure 4.2. The figure showcases two features of emulators: (i) dependence between local accuracy as measured by $s^2(z)$ and grid *size* N_{tr}; and (ii) dependence between $s^2(z)$ and grid *shape*. First, larger training sets improve accuracy (with a general relationship of $\mathcal{O}(N_{tr}^{-1/2})$ like in plain Monte Carlo). This can be seen in Figure 4.2 where kriging standard deviation $s(z)$ is consistently lower for $N_{tr}^{(B)} = 1000$ compared to $N_{tr}^{(A)} = 125$. One implication is that as $N_{tr} \to \infty$, we would have $s^2(z) \to 0$, i.e. $f(\cdot)$ would be learned with

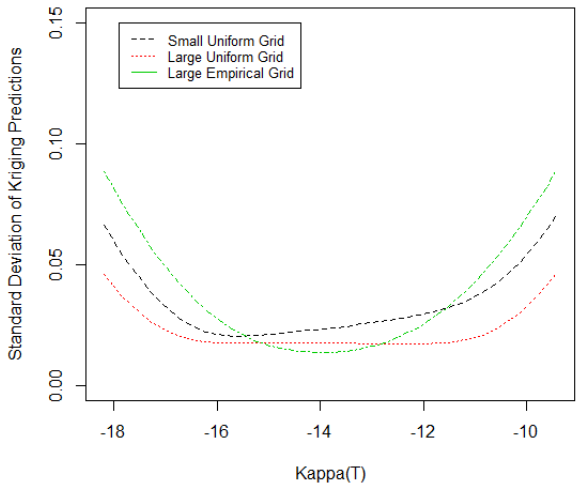

Figure 4.2: Effect of training design $\mathcal{D}$ on the emulator accuracy in the Chen–Cox model. We show kriging standard deviation $s(z)$ for the universal kriging model with three different designs: $\mathcal{D}^{(A)}$ (small uniform), $\mathcal{D}^{(B)}$ (large uniform) and $\mathcal{D}^{(C)}$ (large empirical). The deterministic designs $\mathcal{D}^{(A)}, \mathcal{D}^{(B)}$ contain uniformly spaced values of $\kappa^{(2)}(T) \in (-17.5, 10)$, of size $N_{tr}^{(A)} = 125$ and $N_{tr}^{(B)} = 1000$ respectively. $\mathcal{D}^{(C)}$ is an empirical design of size $N_{tr}^{(C)} = 1000$ generated using the density of $\kappa^{(2)}(T)|\kappa^{(2)}(0)$.

complete precision, a property known as global consistency of the emulator. Second, $s^2(z)$ is affected by the shape of $\mathcal{D}$ in the sense that higher local density of training points lowers the local posterior variance. This is intuitive if viewing $\hat{f}$ as an interpolator or kernel regressor – the denser the training set around z, the better we are able to infer $f(z)$. Consequently, the empirical grid $\mathcal{D}^{(C)}$ that is concentrated around the mode of $Z(T)$, offers better accuracy in that neighborhood (around $\kappa^{(2)}(T) \simeq -14$ in Figure 4.2) compared to the uniform $\mathcal{D}^{(B)}$, but lower accuracy towards the edges, where $\mathcal{D}^{(C)}$ becomes sparser. For all designs, posterior uncertainly deteriorates markedly as we migrate outside of the training

set (e.g. $\kappa^{(2)}(T) > -11$ in the Figure).

The $s(z)$ values shown in Figure 4.2 also provide an approximation of emulator IMSE. For example, averaging the kriging standard deviation $s(z)$ over the testing set using the UK model with $N_{tr} = 1000$ yields $s_{Ave} = \sqrt{\frac{1}{N_{out}} \sum_n s^2(z^{(n)})} = 7.159 \cdot 10^{-3}$, while in Table 4.1 the corresponding reported IMSE was $\widehat{IMSE} = 7.428 \cdot 10^{-4}$. Reasons for the mismatch include the residual MSE in the Monte Carlo estimate $\hat{a}^{MC}$ and model mis-specification of the UK model, which would bias the self-assessed accuracy. Moreover, the strong correlation between $\hat{a}(z)$ across different testing locations $z^{(n)}$ implies that $\widehat{IMSE}$ has a large standard error. Nevertheless, s_{Ave} is a highly useful metric that allows to quantify the relative accuracy of different emulators in the absence of any gold-standard benchmarks.

4.5 Case Study: Hedging an Index-Based Fund in a Two-Population Model

Our second case-study addresses a multi-dimensional setup with four factors. In dimension 4, visualization of the map $z \mapsto a(z, T, x)$ is limited so one must rely on statistical metrics to generate and assess the quality of the emulators.

There has been a lot of recent discussion regarding index-based longevity funds. Information on the death rates of the general public is widely available, and a market fund that uses the respective death rats as its price index offers a standardized way to measure population longevity. In particular, it allows for securitization of longevity swaps that can be used by pension funds to hedge their longevity risk exposure. If the pension fund could buy as many units of the swap as it has to

pay out to its annuitants, it would result in a situation where the amount paid is nearly equal to the amount received from the swap. The quality of such as hedge is driven by the basis risk between the indexed population and the annuitant pool, that is typically a subset of the index. Consequently, it is necessary to create a model to capture the link between the index and the insured sub-population.

Remark. From a different angle, some longevity products explicitly integrate mortality experience in several regions, for example across different countries (UK, Germany, Netherlands) or across different constituencies (England vis-a-vis Great Britain). Lin et al. (2013) state that most mortality data reported by official agencies calculate a weighted average mortality index of different underlying populations. They also investigate the modeling aspect of such multi-population indices.

To fix ideas, we call the index population Pool 1, and the annuitants Pool 2. Consider now an individual from Pool 2 who will be aged x at date T when she begins to receive her life annuity. The corresponding time-T liability to the pension fund is denoted $a_2(Z(T), T, x)$. If the pension fund enters into a swap based on the index, she might purchase π index-fund annuities for age x, with net present value of $\pi a_1(Z(T), T, x)$, at T. For now we ignore what would be a fixed premium. The overall hedge portfolio is then $\Delta(Z(T), T, x) \doteq \pi a_1(Z(T), T, x) - a_2(Z(T), T, x)$. Several risk measures can be used to determine hedge effectiveness. Some examples include variance, or tail risk measures such as value-at-risk (VaR) or expected shortfall (TVaR). Recent work in this direction includes Coughlan et al. (2011) who used a bootstrapping and extrapolation method to analyze hedge effectiveness, and Cairns et al. (2014) whose setup we follow below.

Unsurprisingly, the correlation structure for mortality across populations is complex. One notable recent contribution is by Cairns et al. (2011b, 2014) who considered a hedging problem between an index pool $k = 1$ and insured sub-pool $k = 2$. Specifically, the two populations are the England & Wales (E&W) general population, which represents the index mortality rate (Pool 1), and the Continuous Mortality Investigation (CMI) population, which are mortality rates gathered from United Kingdom insured populations, serving the role of those receiving pension payments (Pool 2). To model the dependence between the two pools, Cairns et al. (2011b) proposed a co-integrated two-population Bayesian model based on the Lee-Carter framework. To wit, the mortality rates $m_k(t, x)$ behave similar to (4.12),

$$\log m_k(t, x) = \beta_k^{(1)}(x) + n_a^{-1}\kappa_k^{(2)}(t) + \frac{1}{n_a}\gamma_k^{(3)}(t - x), \quad k = 1, 2, \tag{4.37}$$

where the subscripts refer to the population index and the superscripts encode the order of the stochastic factors in the manner of Cairns et al. (2011b). The stochastic dynamics of the period effect $\kappa_1^{(2)}$ are given by

$$\kappa_1^{(2)}(t) = \kappa_1^{(2)}(t - 1) + \mu_1 + \sigma_1\epsilon_1(t), \quad \epsilon_1(t) \overset{\text{i.i.d}}{\sim} N(0, 1). \tag{4.38}$$

In turn, the mortality of the larger population influences the period effect of the smaller (insured) population, with dynamics for $\kappa_2^{(2)}$ co-integrated with $\kappa_1^{(2)}$.

Namely, their difference $S(t) \doteq \kappa_1^{(2)}(t) - \kappa_2^{(2)}(t)$ forms an $AR(1)$ process

$$S(t) = \mu_2 + \phi(S(t-1) - \mu_2) + \sigma_2\epsilon_2(t-1) + c\epsilon_1(t-1), \quad \epsilon_2(t) \overset{\text{iid}}{\sim} N(0,1), \tag{4.39}$$

with $\epsilon_1(\cdot)$ independent of $\epsilon_2(\cdot)$, and covariance parameter $c = \sigma_1 - \rho\sigma_2$, where $\rho = \text{Corr}(\kappa_1^{(2)}(t), \kappa_2^{(2)}(t))$. In both models cohort effects $\gamma_k^{(3)}$ are independent $AR(2)$ processes. Since (4.39) models the difference $\kappa_1^{(2)}(t) - \kappa_2^{(2)}(t)$, the mean reversion rate ϕ reflects the rapidity of $S(t)$ returning to its base level μ_2, which is assumed to be the stationary mortality spread between the two populations.

4.5.1 Analytic Approximations

Cairns et al. (2014) used the fact that $\mathbb{E}[\kappa_1^{(2)}(T+t) \mid \kappa_1^{(2)}(T)] = \kappa_1^{(2)}(T) + \mu_1 t$ to introduce the median-mortality approximation

$$\widehat{m}_1^{A1}(T+t, x+t) = \exp\left(\beta_1^{(1)}(x+t) + \frac{1}{n_a}(\kappa_1^{(2)}(T) + \mu_1 t) + \frac{1}{n_a}\gamma_1^{(3)}(T-x)\right). \tag{4.40}$$

Since $S(t)$ is mean reverting, Cairns et al. (2014) also suggested to approximate the CMI population mortality via

$$\widehat{m}_2^{A1}(T+t, x+t) = \exp\left(\beta_2^{(1)}(x+t) + \frac{1}{n_a}(\kappa_2^{(2)}(T) + \mu_1 t) + \frac{1}{n_a}\gamma_2^{(3)}(T-x)\right), \tag{4.41}$$

i.e. the same drift as the general population but different initial value.

We introduce a different, more accurate approximation based on the following lemma.

Lemma 2. *We have*

$$\mathbb{E}\left[\kappa_2^{(2)}(T+t)|Z(t)\right] = \kappa_1^{(2)}(T) + \mu_1 t - \mu_2(1-\phi^t) - \phi^t(\kappa_1^{(2)}(T) - \kappa_2^{(2)}(T)). \tag{4.42}$$

The proof can be found in 4.10. Denote $\mathbb{E}[\kappa_2^{(2)}(T+t)|Z(t)] \doteq \xi(t,T)$. Lemma 2 suggests an alternative analytic estimator for $m_2(T+s,x)$ as

$$\widehat{m}_2^{A2}(T+s,x+s) \doteq \exp\left(\beta_2^{(1)}(x+s) + \frac{1}{n_a}\xi(t,T) + \frac{1}{n_a}\gamma_2(T-x)\right). \tag{4.43}$$

Denote $a_1(Z(T))$ and $a_2(Z(T))$ as the net present value at T (conditional on $Z(T)$) of a life annuity for the E&W and CMI populations respectively as defined in (4.9). In what follows, *Analytic 1* will refer to use of (4.40) and (4.41) in estimating survival probabilities (4.11) for each population (and hence a_1 and a_2), while *Analytic 2* refers to the use of (4.40) and (4.43). Notation for deferred annuity values under the two analytic approaches will be $\hat{a}_k^{A1}(z)$ and $\hat{a}_k^{A2}(z), k = 1, 2$.

4.5.2 Model Fitting

The parameters $\beta_1^{(1)}(x)$, $\beta_2^{(1)}(x)$, and past trajectories $\kappa_k^{(2)}(t)$, $\gamma_k^{(3)}(t-x)$, for $k = 1, 2$ were estimated from the male E&W and CMI populations respectively, and the time and age ranging from calendar years 1961 to 2005 (with 2005 treated as $t = 0$), and x from 50 to 89. The processes $(\kappa_1^{(2)}(t))$ and $(S(t))$ were fit as random walk with drift and $AR(1)$ respectively, introducing additional parameter estimates for $\mu_1, \sigma_1, \mu_2, \phi, \sigma_2$ and c. We find $\mu_1 = -0.5504, \mu_2 = 0.6105, \sigma_1 =$

$1.278, \sigma_2 = 0.568, \phi = 0.9407, c = 0.262$, so that the CMI population tends to have higher mortality $\mu_2 > 0$, with a co-integration of about $\phi = 94\%$.

Using the PPC approach of Cairns et al. (2014), we treat the age-effect parameters as fixed, and refit the ARIMA models at T for each simulation. That is, the $\beta_k^{(1)}$'s are fixed throughout for $k = 1, 2$ and each of μ_1, σ_2, μ_2, ϕ, σ_2, and c are re-estimated. In principle, this makes the re-estimated parameters part of the state $Z(T)$. A few preliminary runs indicate that the variance parameters σ_1, σ_2 and c have little significant effect on annuity values, while μ_1, μ_2 and ϕ do. Since μ_1 is in one-to-one correspondence with $\kappa_1^{(2)}(T)$, our time T state process is finally characterized as

$$Z(T) = \{\kappa_1^{(2)}(T), \kappa_2^{(2)}(T), \mu_2, \phi\}.$$

Heuristically, this is a reasonable choice: each element of $Z(T)$ has a direct effect on the time T mortality rates or their trends, while the variance terms simply add variability.

Several stochastic mortality models have `R` code available[2] for model fitting. We use the LifeMetrics code to fit the two-population model parameters, yielding the inferred past trajectories for the age, period, and cohort effects. In a separate step, the estimated period and cohort effects are modeled as individual ARIMA models.

For the remainder of this section we assume the starting age of the annuitant is $x = 65$ with a fixed interest rate of $r = 0.04$ and a $T = 10$ year deferral period. Generally the hedge ratio π is chosen endogenously, for example through

[2]LifeMetrics Open Source R code for Stochastic Mortality modeling; see `http://www.macs.hw.ac.uk/~andrewc/lifemetrics/` for details

minimizing variance. In this paper we assume the neutral value of $\pi = 1$ in order to not favor one estimation type over another. Hence the value of the hedge portfolio is simply $\Delta(Z(T)) = a_1(Z(T)) - a_2(Z(T))$. Under this setup, the Monte Carlo benchmark yielded an average value of $\mathbb{E}[\Delta(Z(T))] = 0.1995$ with a standard deviation of 0.1067. This suggests that a one-to-one purchase of index annuity is not the optimal hedge ratio under this population model.

As discussed in Section 4.3.4, determining the training set design depends on the problem at hand. In our particular example with a 4-dim. Z, we aim to give an accurate result of the expectation of the hedge portfolio $\Delta(T)$, so we use an empirical design, as suggested in Section 4.3.4. This also holds the advantage of capturing the correlation between $\kappa^{(1)}$ and $\kappa^{(2)}$ which is important in this co-integrated model. To compare the effect of budget size, we choose two different budgets, $N_{tr} = 1000$ and $N_{tr} = 8000$. Following the framework in Section 4.3.4, N_{tr} is allocated into $N_{tr,1} = N_{tr}^{2/3}, N_{tr,2} = N_{tr}^{1/3}$, so that we have $N_{tr,1} = 100$ (resp. $N_{tr,1} = 400$) training points with Monte Carlo simulations containing $N_{tr,2} = 10$ (resp. $N_{tr,2} = 20$) batched simulations for each design point.

Different surrogate models are chosen than in Section 4.4; this time around a multi-dimensional state process suggests the use of a spline (namely TPS) model from Section 4.3.1. We forego the OK model, but maintain use of the 1st-order linear UK model, and also implement a simple kriging (SK) model with trend $\mu(z) = \hat{a}_1^{A2}(z) - \hat{a}_2^{A2}(z)$. This combines advantages from both the analytic and UK approach, giving us an already accurate estimate for the trend, while non-parametrically modeling the residuals. For these reasons, a SK emulator should outperform both the analytic estimators and the UK model. We utilize another

advantage of the surrogate models and fit them directly to the hedge portfolio values $\Delta(Z(T))$ rather than individually modeling annuity values $a_k(Z(T))$ and then taking difference of two approximations. Additionally, we implement the Least Square Monte Carlo method using a linear combination of all polynomials of degree 2 or less in z as another comparison tool. Since the dimension of $Z(T)$ is $d = 4$, this leads to $p = d^2 = 16$ basis functions which is reasonable given the relatively small values of N_{tr}.

4.5.3 Results

	$N_{tr} = 1000$		$N_{tr} = 8000$	
Type	Bias	$\sqrt{\text{IMSE}}$	Bias	$\sqrt{\text{IMSE}}$
Analytic A1 from (4.41)	-2.101e-02	3.460e-02	-2.101e-02	3.460e-02
Analytic A2 from (4.43)	3.629e-03	3.733e-03	3.629e-03	3.733e-03
Thin Plate Spline	-1.050e-03	1.437e-02	4.431e-04	3.294e-03
Universal Kriging	-1.156e-03	1.872e-02	2.556e-03	1.454e-02
Simple Kriging	2.148e-03	2.308e-03	9.229e-04	1.469e-03
Least Squares MC	-1.050e-03	1.437e-02	5.324e-04	3.295e-03

Table 4.2: Performance of analytic estimates and surrogate models for hedge portfolio values in the two-population model case study. Numbers reported are based on $N_{out} = 1000$ simulations of $Z(T)$ with a Monte Carlo benchmark. N_{tr} is allocated into $N_{tr,1} = N_{tr}^{2/3}$ training points and $N_{tr,2} = N_{tr}^{1/3}$ Monte Carlo batches per training point. Simple kriging model uses A2 estimator as trend. For comparison purposes, the average value of the hedge portfolio was 0.1995.

We choose $N_{out} = 1000$ simulations of $Z(T)$ and predict hedge portfolio values $\Delta(Z(T))$ with the surrogate models, as well as via the deterministic estimates. Table 4.2 shows the results. As expected, the Analytic A2 estimator outperforms Analytic A1 since it is catered directly to the two-population model. Relative to A1, our improved estimator cuts bias by nearly 80%. As for the surrogates,

when $N_{tr} = 1000$, each of the TPS and UK models only slightly underperform the analytic estimate A2, while the SK model does significantly better. For $N_{tr} = 8000$, both TPS and SK are better than A2. The LSMC and TPS values are equal for $N_{tr} = 1000$ and very close for $N_{tr} = 8000$. This is unsurprising given Equation (4.17), which shows that the spline behaves similarly to least squares. In our case, both the LSMC and TPS methods used second order polynomial basis functions, and the sparse grid for $N_{tr} = 1000$ resulted in a large enough smoothing parameter λ to be insignificant relative to the estimate up to four decimals.

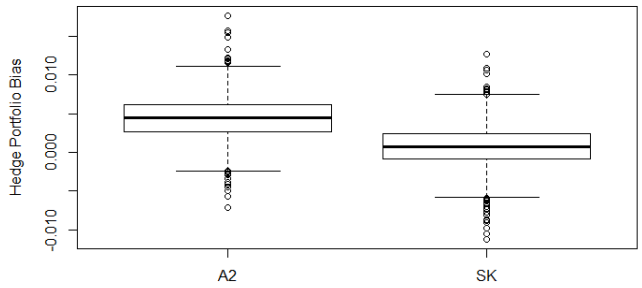

Figure 4.3: Boxplots of hedge portfolio value bias for $N_{tr} = 8000$ for analytic A2 and simple kriging approaches. To construct the boxplot, we computed for each of 1000 simulated values of $Z(T)$, the difference between the respective estimate and the Monte Carlo benchmark.

Figure 4.3 summarizes the empirical distribution of the bias of the A2 and SK estimators given simulations of $Z(T)$. We can see that both approaches have similar variability, while SK has a much lower bias. The UK and TPS estimators have similar distributions with slightly larger bias than SK.

There are a few comments to be made in regards to these results. First of all, there is no way to tell *a priori* that a deterministic estimate will perform well. For example each surrogate model completely outclasses A1, while TPS and

UK perform only marginally better than A2. Possibly, even better (or worse) analytic estimators can be derived. Additionally, the deterministic estimators are for annuity values themselves and not for the portfolio difference $\Delta(T)$. A lower bias for a $\Delta(T)$ could simply be a consequence of the bias of each annuity $a_k(Z(T))$ being canceled during subtraction. We also mention that in terms of percentage errors these biases are significant, being on the order of 1% of the benchmarked $\hat{a}^{MC} = 0.1995$.

4.6 Case Study: Predicting Annuity Values under the CBD Framework

4.6.1 Model Fitting

Our third case study utilizes another popular class of mortality models, the CBD Cairns et al. (2006) models which directly work with the survival probabilities. To wit, we model the 1-period survival probability

$$P(Z(T); T, 1, x) = \frac{1}{1 + \exp\left(\kappa^{(1)}(T) + (x - x_{Ave})\kappa^{(2)}(T)\right)}, \tag{4.44}$$

where $x_{Ave} = n_a^{-1} \sum_i x_i$, and $\kappa^{(1)}, \kappa^{(2)}$ follow ARIMA models, which according to Cairns et al. (2011a) provide a good fit for period effects. Multi-period survival probabilities are obtained as products of (4.44).

We fit (4.44) to the CMI population, considering a full range of $ARIMA(p, d, q)$ models with $p, q = 0, 1, 2, 3, 4$ and $d = 0, 1, 2$, using `auto.arima` in `R` from the package "forecast" Hyndman (2015). The optimal configuration for this population is

for $\kappa^{(1)}$ to follow $ARIMA(0,1,3)$ with drift and $\kappa^{(2)}$ to follow $ARIMA(1,1,2)$:

$$\kappa^{(1)}(t) = \kappa^{(1)}(t-1) + \mu + \epsilon^{(1)}(t) + \sum_{q=1}^{3} \theta^{(q,1)} \epsilon^{(1)}(t-q), \tag{4.45}$$

$$\kappa^{(2)}(t) = (1+\phi)\kappa^{(2)}(t-1) - \phi\kappa^{(2)}(t-2) + \epsilon^{(2)}(t) + \sum_{q=1}^{2} \theta^{(q,2)} \epsilon^{(2)}(t-q). \tag{4.46}$$

The estimated ARIMA parameters are $\mu = -0.0195, \phi = 0.9206, \theta^{(1,1)} = -0.5516, \theta^{(2,1)} = 0.1736, \theta^{(3,1)} = 0.5169, \theta^{(1,2)} = -1.4664, \theta^{(2,2)} = 0.6167$. The $\theta^{(\cdot,k)}, k = 1, 2$ describe how past errors echo into future values of $\kappa^{(k)}$. For example, the large negative value of $\theta^{(1,2)}$ means that the noise generated in $\kappa^{(2)}(s)$ will be amplified, made negative, and added to the future $\kappa^{(2)}(s+1)$. The above equations imply that the mortality state has three components,

$$Z(T) = \{\kappa^{(1)}(T), \kappa^{(2)}(T), \kappa^{(2)}(T-1)\}.$$

As in the previous case studies, we develop a deterministic estimate for survival probabilities. Denote by $\xi^{(k)}(t,s) \doteq \mathbb{E}[\kappa^{(k)}(t) \mid Z(s)]$ for $k = 1, 2$. The expressions for $\xi^{(k)}$ are as follows.

Lemma 3. *The following hold for $t > s$*

$$\xi^{(1)}(t,s) = \kappa^{(1)}(s) + \mu(t-s); \tag{4.47}$$

$$\xi^{(2)}(t,s) = \phi^{t+1-s} \left(\frac{\kappa^{(2)}(s) - \kappa^{(2)}(s-1)}{\phi - 1} \right) + \left(\frac{\phi\kappa^{(2)}(s-1) - \kappa^{(2)}(s)}{\phi - 1} \right). \tag{4.48}$$

The proof can be found in 4.10.3. Based on Lemma 3, and substituting expected values of $\kappa^{(k)}(s)$ into (4.44) we obtain a deterministic estimate of the u-year

survival probability as the product

$$\widehat{P}^{det}(Z(s),t,u,x)=\prod_{j=0}^{u-1}\frac{1}{1+\exp\left(\xi^{(1)}(t+j,s)+(x+j-x_{Ave})\xi^{(2)}(t+j,s)\right)}.$$

Through equation (4.9), this yields the estimate for the $T-$year deferred annuity given $Z(T)$:

$$\hat{a}^{det}(Z(T),T,x)=\sum_{s=1}^{\bar{x}-x}e^{-rs}\hat{P}(Z(T),T,s,x), \tag{4.49}$$

where the cutoff age is $\bar{x}=89$.

We proceed to value life annuities in the above model. In contrast to the first two case studies, we extend the deferral period to twenty years. An additional ten years of evolution imbues significant uncertainty into the mortality state $Z(T)$. We use an empirical training design $\mathcal{D}$ for this case study for two reasons; one being that the correlation structure among the components of $Z(T)$ is problematic with any rectangular grid. From (4.46), we see that $\kappa^{(2)}(20)$ and $\kappa^{(2)}(19)$ should be strongly correlated, while both $\kappa^{(2)}(19)$ and $\kappa^{(2)}(20)$ are independent of $\kappa^{(1)}(20)$. Secondly, the long deferral period causes significant variation in the distribution of $Z(20)$, and with expectation in mind, we desire the empirical grid's ability to accurately capture the density of $Z(T)$. The algorithms discussed in Sections 4.3.4 and 4.3.5 are used to generate the design and fit the surrogate models. As in Section 4.4, we choose an ordinary kriging and 1st-order linear universal kriging models, and also fit a thin plate spline model as used in Section 4.5.

4.6.2 Results

	$N_{tr} = 1000$		$N_{tr} = 8000$	
Type	Bias	$\sqrt{\text{IMSE}}$	Bias	$\sqrt{\text{IMSE}}$
Analytic from (4.49)	-4.560e-01	5.257e-01	-4.560e-01	5.257e-01
Thin Plate Spline	-2.358e-02	6.719e-02	4.195e-03	5.436e-02
Ordinary Kriging	3.669e-03	9.785e-02	9.734e-03	7.743e-02
Universal Kriging	-1.785e-03	5.844e-02	5.635e-03	4.355e-02

Table 4.3: Performance of analytic estimates and surrogate models for 20-year deferred annuity values under the CBD framework. Numbers reported are based on $N_{out} = 1000$ draws of $Z(20)$. N_{tr} is allocated into $N_{tr,1} = N_{tr}^{2/3}$ training points and $N_{tr,2} = N_{tr}^{1/3}$ Monte Carlo batches per training point. Analytic estimate refers to $\hat{a}^{det}$ in (4.49). Universal kriging model uses linear basis functions.

In contrast to the results in Sections 4.4.1 and 4.5.3, Table 4.3 shows that the analytic estimator (4.49) crumbles under this volatile model and long deferral period. On the other hand, both kriging models produce reasonable results even with $N_{tr} = 1000$. We can also observe a diminished effect of increasing the training set size, due to the increased model variance.

These results reflect the comments made in the previous sections: the analytic estimate is a parametric guess as to what may provide an accurate result, and that guess is not always correct. Our analytic choice in this case study was derived along identical lines as to the analytic estimates in the other case studies, yet performs substantially worse. In comparison, the statistical learning frameworks provide a reliable estimator even in a volatile model with a three-dimensional state process and long deferral period.

4.7 Case Study: Valuation of Equity-Indexed Annuities

Our last case study moves beyond fixed annuities to a more complex variable annuity. We consider a product with equity-linked payments in terms of returns on an index ($S(t)$), as well as stochastic interest rates ($r(t)$).

To highlight the flexibility of the emulation framework, we discuss modeling one-year forward values for an *annual reset* EIA. In such a contract (see Lin and Tan (2003); Qian et al. (2010) for an overview), the buyer is guaranteed a minimum rate of return g. In addition, she is entitled to higher payments if the annual return $R(t) = S(t)/S(t-1) - 1$ of the index S exceeds g; this upside has a participation rate $\alpha < 1$. Following Lin and Tan (2003), we assume the annuitant is aged x and holds the EIA until expiration time T, interpreted as time of retirement, whence the fund is converted into a traditional fixed annuity. If death happens before T, then the current fund value is paid at the end of year of death. Let $r(t)$ be instantaneous interest rate at time t, and $\tau(x)$ be the remaining lifetime of the individual. Conditioning on $\tau > t$ and the information $\mathcal{F}(t)$ available by time t, and assuming that accrued payments by t are normalized to be 1, the present value of this EIA at t is

$$f(Z(t)) \doteq \sum_{j=t}^{T-1} \mathbb{E}\left[e^{-\int_t^{j+1} r(u)du} \prod_{i=t}^{j+1} \max\left(e^{\alpha R(i)}, e^{g}\right) 1_{\{j<\tau(x)\leq j+1\}} \middle| \mathcal{F}(t) \right] + \mathbb{E}\left[e^{-\int_t^{T} r(u)du} \prod_{i=t}^{T} \max\left(e^{\alpha R(i)}, e^{g}\right) 1_{\{\tau(x)>T\}} \middle| \mathcal{F}(t) \right], \tag{4.50}$$

Denote by $C(t,s)$ the expected payoff at period s discounted to $t \leq s$:

$$C(t,s) \doteq \mathbb{E}\left[e^{-\int_t^s r(u)du} \prod_{i=t}^{s} \max\left(e^{\alpha R(i)}, e^{g}\right) \middle| \mathcal{F}(t)\right], \tag{4.51}$$

Assuming independence of mortality dynamics from the financial quantities, we can re-write (4.50) as

$$f(Z(t)) = \sum_{j=t}^{T-1} C(t,j+1)\left[P(Z(t);t,j,x) - P(Z(t);t,j+1,x)\right] + C(t,T)P(Z(t);t,T,x), \tag{4.52}$$

where $P(Z(t);t,j,x) - P(Z(t);t,j+1,x)$ is the expected probability of death in year j given the initial mortality state $Z(t)$. Determining the map $Z(t) \mapsto f(Z(t))$ is important for risk measure analysis, such as under Solvency II which requires knowledge of the liabilities distribution at a future point in time.

To precise the modeling of $(Z(t))$ we blend the setups of Qian et al. (2010) and Cairns et al. (2011a). All financial assets are specified under the risk-neutral measure $\mathbb{Q}$ with spot interest rates following the Cox-Ingersoll-Ross model

$$dr(t) = \gamma\left(\beta - r(t)\right)dt + \sigma_r\sqrt{r(t)}dW_r(t), \tag{4.53}$$

and the risk index S following Geometric Brownian motion:

$$d\log S(t) = \left(r(t) - \frac{1}{2}\sigma_S^2\right)dt + \sigma_S dW_S(t), \tag{4.54}$$

where W_r and W_S are independent standard Brownian motions. Note that this

implies that the $\mathbb{Q}$-distribution of the returns $R(t) = S(t)/S(t-1) - 1$ depends on the interest rates $\{r(s) : t-1 < s < t\}$. It also shows that in fact $S(t)$ is not part of $Z(t)$, since the distribution of $R(t)$ is independent of the index level, only its increments.

For the mortality model, we choose the Lee-Carter model with cohort effect as in (4.37) and (4.38), fitted to E&W general population data as described in Section 4.5. The end result is a two-dim. state process $Z(t) = \{r(t), \kappa(t)\}$. Since the dynamics of $\kappa(\cdot)$ are independent from those of $r(\cdot)$, one could in principle build separate emulators for each, which could allow to re-use emulation for other EIA's (or other interest-rate-sensitive products).

4.7.1 Results

For the remainder of this section we analyze $f(Z(1))$, the net present value of the EIA one year into the future. We let $x = 55$ and the expiration time be $T = 10$. Our guaranteed rate is $g = 0.03$, the CIR model (4.53) has parameters $r(0) = 0.04, \gamma = 0.6, \beta = 0.04, \sigma_r = 0.03$, from Qian et al. (2010), and the mutual fund has volatility $\sigma_S = 0.2$. Following Lin and Tan (2003), we set the participation index α to solve $f(Z(0)) = 1$, resulting in $\alpha = 0.8211$.

We proceed to fit kriging and LSMC emulators, using an empirical training design $\mathcal{D}$. We employ linear basis functions for LSMC and a first-order linear trend model for Universal Kriging. The algorithms discussed in Sections 4.3.4 and 4.3.5 are used to generate the design and fit the surrogate models. Because $(r(t))$ and $(R(t))$ are specified through a continuous-time model, we employ a standard Euler method with time-step $\Delta t = 0.01$ to simulate interest rates and stock returns on

$[t, T]$. This procedure makes simulation much more expensive (about two orders of magnitude slower than previous examples) and hints at speed advantages of emulation over plain nested Monte Carlo. The emulation budget is $N_{tr} = 10,000$ split into $N_{tr,1} = 100$ outer design points and $N_{tr,2} = 100$ for the batch size. The Monte Carlo benchmark is generated based on a grid of $N_{out} = 25$ values of $r(1)$, with $N_{in} = 10^4$ and final result smoothed by a spline.

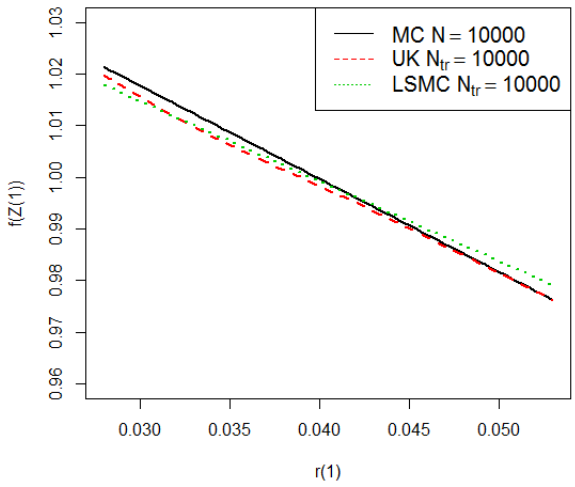

Figure 4.4: Marginal dependence plot of $f(Z(1))$ versus $r(1)$, where $f(Z(1))$ is defined in (4.52) and is estimated through a smoothed Monte Carlo benchmark with $N_{out} = 25, N_{in} = 10,000$. The two emulator models used $N_{tr,1} = N_{tr,2} = 100$. The 2-dim. experimental design $\mathcal{D}$ for the emulators was empirical.

While emulators were fit for the full 2-dim. map $Z(1) \mapsto f(Z(1))$, Figure 4.4 shows the partial dependence of the EIA value on $r(1)$ (experiments show that $f(Z(t))$ is much more sensitive to interest rates than to the mortality factor $\kappa(1)$). For the plot we fixed $\kappa(1) = -13.76$ which is the estimated mean of $\kappa(1)$. We

observe a decreasing relationship between $r(1)$ and $f(Z(1))$ similar to a traditional annuity. The emulators capture this link, but the LSMC version has trouble at the edges of the distribution; here the mismatched slope for the linear basis functions causes issues at the edges. We remark that a quadratic fit results in the same issue. To give a numeric indication of the model differences, the root mean squared difference between the model prediction and Monte Carlo benchmark over the 25 test points was 1.649e-03 for the kriging model and 2.023e-03 for LSMC, which is roughly 20% better.

To reiterate the efficiency gains of the emulators, the nested Monte Carlo benchmark (which was only modeling the 1-dim. map $r \mapsto f(r, -13.76)$) took several hours to run on our laptop with total simulation budget of $N = 2.5 \cdot 10^5$; the kriging emulator with $N_{tr} = 10^4$ took a few minutes and the LSMC emulator with same budget was even faster, since it had smaller regression overhead.

4.8 Conclusion

The four case studies above showcase the flexibility and performance of the surrogate models across a range of various longevity risk dynamics and applications. Compared to the consistent accuracy of the statistical emulators, the quality of the deterministic projections was widely varying. Because an analytic derivation is required to produce a deterministic estimator, there are several plausible estimators available. In Section 4.5 we derived two different estimators, both of which were viable, but one underperformed. Similarly, in Section 4.6 the derived deterministic projection was also inaccurate. Overall, these examples show that our

models can outperform deterministic projections and provide minimally-biased estimates. Moreover, despite the fact that sometimes analytic approximation were just as good, we hope that our experiments serve as a cautionary tale on using approximations without quantifying their accuracy.

Our examples show that statistical emulators offer a principled approach to Monte Carlo approximation of annuity present values. By endogenously generating proxies for approximation accuracy, emulators nicely complement analytic formulas. Moreover, through their flexibility, emulators can work even in complicated settings where analytic derivations are hard to come by. On the latter aspect, we emphasize the advantages of working with a non-parametric framework such as stochastic kriging compared to parametric least-squares approaches where the modeler must explicitly specify the structure (i.e. basis functions and their number) of the emulator. We also point out the ability of kriging models to incorporate analytic approximations as trend functions which allows to combine the best of both worlds, see e.g. the excellent performance of the SK model in Section 4.5.

Throughout the paper we have hinted at possibilities for further work. Straightforward extensions include using other mortality models, or emulating other insurance products. For example, Bacinello et al. (2011) provides an in-depth analysis of many types of variable annuities using Least Squares Monte Carlo. Both kriging and spline models could be used in place of the least squares framework to provide an additional viewpoint to the problem. A useful example to analyze is when surrenders and withdrawals are available. One can also build more dynamic surrogates that treat initial age x (fixed in our case studies as $x = 65$)

or deferral period T as part of the state Z, providing a joint prediction for $(Z(0), T, x) \mapsto \mathbb{E}[a(Z(T), T, x)]$. Similarly, one could consider more parameter uncertainty which would lead to including additional components in the state Z.

The emulators we obtained offer a high-performance tool for annuity risk management. Indeed, they are based on advanced, previously vetted stochastic mortality models and calibrated to real, reliable, large-scale mortality datasets. Hence, the fitted estimates for annuity values are in essence a best-available forecast that combines state-of-the-art longevity modeling, data calibration and statistical model. As such, (after incorporating age and interest rate as model parameters) they would be of independent interest to actuaries working in longevity space and seeking easy-to-use tools for forecasting net present values of life annuities. The emulator offers a plug-and-play functionality, converting inputted parameters (such as age x, deferral period T and discount rate r) into the annuity value (note that the initial state $Z(0)$ is read off from the calibration procedure). One can imagine building a library of such emulators for different mortality-contingent products available in the marketplace.

Looking more broadly, the emulation approach we propose is very general and can be applied in a variety of actuarial contexts. In particular, in future work we plan to extend it to the microscopic agent-based models of mortality Barrieu et al. (2012) which offer a canonical "complex system" representation of population longevity. We believe that emulators could significantly simplify predictions in these types of models by providing a tractable, statistical representation of demographic interactions within a stochastic dynamic population framework. Another class of insurance applications requires functional-regression tools where

emulators can again be very effective Gan and Lin (2015). A different extension is emulation of risk measures related to $F(T, Z(\cdot))$, such as VaR or TVaR, which require targeted surrogates that focus on a specific region of the input space. A starting point is to combine concept of importance sampling to generate a targeted design $\mathcal{D}$ that e.g. preferentially concentrates on the left tail of F. Figure 4.4 shows that the nonparametric regression underlying emulation is well-suited to perform tail analysis for any stochastic factor, such as interest rate, that drives risk in insurance products.

Acknowledgments

We thank the Associate Editor and the Anonymous referees for many helpful comments that improved the paper. We also acknowledge the feedback from the participants at the 2015 Actuarial Research Conference and the 2015 Longevity 11 where earlier versions of the article were presented. The research of the first author is partly funded by the SOA Hickman Scholarship.

4.9 Appendix: Lee Carter & CBD Stochastic Mortality Models

In this section we give a brief summary of existing stochastic mortality models. We use the notation of Cairns et al. (2011a) who provided a comprehensive comparison of several mortality models using CMI data.

The APC Lee-Carter model (introduced by Renshaw and Haberman (2006)) models the log mortality rate as

$$\log m(t,x) = \beta^{(1)}(x) + \beta^{(2)}(x)\kappa^{(2)}(t) + \beta^{(3)}(x)\gamma^{(3)}(t-x). \tag{M2}$$

One can interpret $\beta^{(1)}(x)$, $\kappa^{(2)}(t)$ and $\gamma^{(3)}$ as the age, period and cohort effects, respectively. The original model proposed by Lee and Carter (1992) is a special case where $\gamma^{(3)} = 0$. The age effects $\beta^{(k)}(x), k = 1, 2, 3$ are estimated (non-parameterically) from historical data, while the period and cohort effects are taken as stochastic processes. In the original proposal in Lee and Carter (1992), the period effect $\kappa^{(2)}$ is assumed to follow a random walk (i.e. unit root $AR(1)$ in discrete time),

$$\kappa^{(2)}(t) = \kappa^{(2)}(t-1) + \mu^{(2)} + \sigma^{(2)}\epsilon^{(2)},$$

where $\mu^{(2)}$ is the drift, $\sigma^{(2)}$ is the volatility, and $\epsilon^{(2)} \sim N(0,1)$ i.i.d. is the noise term. Alternatively, Cairns et al. (2011a) mention that $ARIMA$ models may provide a better fit, in particular fitting an $ARIMA(1,1,0)$ process for $\kappa^{(2)}$ based on 2007 CMI dataset.

For the cohort effect, Renshaw and Haberman (2006) suggested using $ARIMA$

models for $\gamma^{(3)}(t-x)$; Cairns et al. (2011a) recommend the use of either $ARIMA(0,2,1)$ or $ARIMA(1,1,0)$. Renshaw and Haberman (2006) and Cairns et al. (2011a) both assume $\gamma^{(3)}$ is independent of $\kappa^{(2)}$.

This model has identifiability issues, and one set of constraints could be

$$\sum_t \kappa^{(2)}(t) = 0, \quad \sum_x \beta^{(2)}(x) = 0, \quad \sum_{x,t} \gamma^{(3)}(t-x) = 0, \quad \text{and} \quad \sum_x \beta^{(3)}(x) = 1.$$

From a different perspective, Cairns, Blake, and Dowd (2006) (CBD) proposed a model for $q(t,x) = 1 - P(Z(0); t, 1, x)$, the probability of death in year t for someone aged x. Namely, they use

$$\operatorname{logit} q(t,x) = \beta^{(1)}(x)\kappa^{(1)}(t) + \beta^{(2)}(x)\kappa^{(2)}(t), \tag{M5}$$

where $\operatorname{logit}(y) = \log\left(\frac{y}{1-y}\right)$.

If we let n_a be the number of ages available in the data set for fitting, and take $x_{Ave} = n_a^{-1}\sum_i x_i$, the commonly used parameterization for the CBD model (M5) is

$$\beta^{(1)}(x) = 1, \qquad \text{and} \quad \beta^{(2)}(x) = x - x_{Ave}. \tag{4.55}$$

Under these assumptions there are no identifiability issues.

4.10 Appendix: Proofs of Analytic Estimates

4.10.1 Proof of Lemma 1.

Since the noise terms $\xi^{(k)}(u)$ are independent of $\kappa^{(2)}(s)$ for $u \neq s$, taking conditional expectation with respect to $Z(s) = \{\kappa^{(2)}(s), \xi^{(2)}(s)\}$, and writing in terms of the increments $\kappa^{(2)}(u) - \kappa^{(2)}(u-1)$ yields

$$\begin{aligned}\mathbb{E}\left[\kappa^{(2)}(t) - \kappa^{(2)}(s) \mid Z(s)\right] &= \sum_{u=s+1}^{t} \mathbb{E}\left[\kappa^{(2)}(u) - \kappa^{(2)}(u-1) \mid Z(s)\right] \\ &= \sum_{u=s+1}^{t} \mathbb{E}\left[\xi^{(1)}(u) + \xi^{(2)}(u) - \xi^{(2)}(u-1) \mid Z(s)\right].\end{aligned} \tag{4.56}$$

By the independence assumption we have for $u \neq s+1$

$$\mathbb{E}\left[\xi^{(1)}(u) \mid Z(s)\right] = \mu^{(1)} \tag{4.57}$$

$$\mathbb{E}\left[\xi^{(2)}(u) - \xi^{(2)}(u-1) \mid Z(s)\right] = \mu^{(2)}p - \mu^{(2)}p = 0. \tag{4.58}$$

For $u = s+1$,

$$\mathbb{E}\left[\xi^{(2)}(s+1) - \xi^{(2)}(s) \mid Z(s)\right] = \mu^{(2)}p - \xi^{(2)}(s). \tag{4.59}$$

Combining (4.56)-(4.59), we obtain

$$\mathbb{E}\left[\kappa^{(2)}(t) \mid Z(s)\right] = \kappa^{(2)}(s) + (t-s)\mu^{(1)} + \mu^{(2)}p - \xi^{(2)}(s). \tag{4.60}$$

□

4.10.2 Proof of Lemma 2.

Since κ_1 has trend μ_1, $\mathbb{E}[\kappa_1(t) - \kappa_1(t-1)] = \mu_1$, and using conditional independence, we obtain,

$$\mathbb{E}\left[\kappa_1(T+t) \mid Z(T)\right] = \kappa_1(T) + \mu_1 t. \tag{4.61}$$

For the co-integration term $S(t)$, the expected values satisfy

$$\mathbb{E}[S(T+t) \mid Z(T)] = \mu_2 + \phi\left(\mathbb{E}[S(T+t-1) \mid S(T)] - \mu_2\right). \tag{4.62}$$

The above gives a recursive equation for $t \mapsto \mathbb{E}[S(T+t) \mid Z(T)]$, with initial condition $\mathbb{E}[S(T+0) \mid Z(T)] = S(T)$, which can be solved to yield

$$\mathbb{E}[S(T+t) \mid Z(T)] = \mu_2(1-\phi^t) + \phi^t S(T). \tag{4.63}$$

Finally, using $\kappa_2(t) = \kappa_1(t) - S(t)$, and combining (4.61) with (4.63) leads to

$$\mathbb{E}[\kappa_2(T+t) \mid Z(T)] = \kappa_1(T) + \mu_1 t - \left(\mu_2(1-\phi^t) + \phi^t\left[\kappa_1(T) - \kappa_2(T)\right]\right).$$

as desired. □

4.10.3 Proof of Lemma 3

For $\xi^{(1)}(t,s)$, $\kappa^{(1)}$ is no different than a random walk with drift, so we have

$$\mathbb{E}[\kappa^{(1)}(t) \mid \kappa^{(1)}(s)] = \kappa^{(1)}(s) + \mu(t-s), \quad s \leq t.$$

Next, we take expectation on both sides of (4.46) to obtain the recursive relation

$$\mathbb{E}[\kappa^{(2)}(t) \mid Z(s)] = (1+\phi)\mathbb{E}[\kappa^{(2)}(t-1) \mid Z(s)] - \phi\mathbb{E}[\kappa^{(2)}(t-2) \mid Z(s)] \quad (4.64)$$

where $Z(s) = \{\kappa^{(1)}(s), \kappa^{(2)}(s), \kappa^{(2)}(s-1)\}$. Equation (4.64) is a recursive relation in t with general solution

$$\mathbb{E}[\kappa^{(2)}(t) \mid Z(s)] = c_1\phi^t + c_2, \quad (4.65)$$

where the constants c_1 and c_2 are to be determined. Plugging-in the initial conditions

$$c_1\phi^s + c_2 = \mathbb{E}[\kappa^{(2)}(s) \mid Z(s)] = \kappa^{(2)}(s), \quad \text{and} \quad (4.66)$$

$$\begin{aligned} c_1\phi^{s+1} + c_2 &= \mathbb{E}[\kappa^{(2)}(s+1) \mid Z(s)] = (1+\phi)\mathbb{E}[\kappa^{(2)}(s) - \phi\kappa^{(2)}(s-1) \mid Z(s)] \\ &= (1+\phi)\kappa^{(2)}(s) - \phi\kappa^{(2)}(s-1). \end{aligned} \quad (4.67)$$

and solving for c_1, c_2 we obtain

$$c_1 = \phi^{1-s}\frac{\kappa^{(2)}(s) - \kappa^{(2)}(s-1)}{\phi - 1}, \qquad c_2 = \frac{\phi\kappa^{(2)}(s-1) - \kappa^{(2)}(s)}{\phi - 1}. \quad (4.68)$$

Finally, combining (4.68) with (4.65), we arrive at (4.48). □

Chapter 5

Sequential Design Algorithms for Estimating Value-At-Risk for Longevity Risk

5.1 Introduction

The latest insurance and financial regulators mandate estimation of quantile-based portfolio risk measures. The Solvency II Christiansen and Niemeyer (2014) framework calls for computing the 99.5%-level *value-at-risk* for a 1-year horizon, while the Basel 3 regulations in banking Committee et al. (2013), requires to report the $\alpha-$level *tail value-at-risk* (TVaR_α)). These quantities are typically calculated by first generating a representative set $\mathcal{Z}$ of future loss scenarios and then evaluating the empirical quantile based on $\mathcal{Z}$. However, due to the underlying cashflow and valuation complexity, directly evaluating future portfolio loss is usually not feasible and instead approximate losses are computed. This is frequently done by a Monte Carlo evaluation of the corresponding conditional expectation, leading to *nested* simulation problem.

In the plain nested Monte Carlo method, for each outer scenario one simply computes future portfolio losses using a fixed number of inner scenarios, averages the losses, and takes the quantile or tail average of these results. This often becomes computationally infeasible, so that practitioners sometimes rely on crude approximations that avoid/minimize inner simulations but introduce errors due to misspecification Gordy and Juneja (2010). Furthermore, these errors are difficult to quantify and in fact such empirical VaR_α estimate are necessarily biased.

In practice, the scenarios $\mathcal{Z}$ are realizations or samples based on some underlying stochastic factors or risk drivers (Z_t). Thus, we can identify each $z \in \mathcal{Z}$ as the value of $Z_T = z$, and the portfolio loss can be abstractly viewed as evaluating the expected cashflow Y (which depends on the future path $(Z_t)_{t \geq T}$) given the initial condition $Z_T = z$.

$$f(z) \equiv \mathbb{E}\left[Y\left((Z_t)_{t \geq T}\right) | Z_T = z\right]. \tag{5.1}$$

Note that we assume that Z is Markovian which is essentially always the case in the practical context; if necessary Z is augmented to make it Markov. We assume that $f(z)$ is not available in closed form, so it must be approximated via *inner* step of the nested procedure. Our goal are efficient and generic ways of estimating VaR_α and TVaR_α when the risk drivers follow a simulatable Markov process.

The key idea is to improve the *inner* step of the nested procedure by adaptively allocating simulation budget to scenarios with large losses. Indeed with $\alpha = 99.5\%$, roughly 99% of the outer scenarios are irrelevant from the point of view of VaR or TVaR computation. To maximize the adaptive procedure, we employ

statistical emulation which treats f as an unknown function and seeks to produce a function estimate $\hat{f}$. The underlying idea is that nearby inputs should produce similar outputs, so from a Bayesian perspective, we can make inference on $f(z)$ for *any* z simply based on what was already simulated at its neighbors. The typical modeling method used in emulation is *Gaussian process (GP) regression*, or *kriging* Rasmussen and Williams (2006) (also known as *stochastic kriging* when there is intrinsic uncertainty inherited from stochastic simulation). With this, the posterior output $\hat{f}$ is multivariate Gaussian, with its distribution based on the y_i and s_i at nearby scenarios. Hence any statistical quantities (e.g. quantiles, posterior variance) are easily computable.

Remark (Emulation versus Regression). Emulation is quite similar to regression. Two conceptual differences is that (i) with regression the dataset (z_i, y_i) is given a priori to the modeler, and the goal is find the relationship between y's and $z's$. With emulation, the modeler is in charge of the simulations which are used to *learn* the input-output pairing as quickly as possible. Furthermore, (ii) with regression one can often "see" the shape of f and the noise, and so a parametric approach is a sensible to refine this "shape"; with emulation the data come in on-the-fly and hence usually sequential/non-parametric paradigms are preferred which can automatically infer the latent relationship.

In the setting of approximating a black-box function like in Equation (5.1), emulation has already proven its efficiency in various settings, see e.g. Santner et al. (2013), Fang et al. (2005), Rasmussen and Williams (2006), Forrester et al. (2008), Gramacy and Lee (2008), Marrel et al. (2008). More specifically, stochastic kriging has been used numerously in financial settings. Closely related to this

paper is Risk and Ludkovski (2016) that studied approximate pricing of longevity-linked contracts, requiring an average of Equation (5.1) over the whole distribution Z_T. However, when the quantity of interest is based on tail risk, the problem is inherently different, since the fitted $\hat{f}$ only needs to be accurate in the tail of $f(Z_T)$.

Despite an expansive literature on related topics, we have found no direct applications to the problem of VaR_α (or any exact quantile) estimation in the Monte Carlo setting (5.1). There is, however, a vast collection of supplementary ideas using emulation. We briefly list some fundamental results here, with detailed discussion later:

- Oakley (2004) was the seminal work on emulation and kriging for quantile estimation of expensive computer code output (non-stochastic). Used is a three-stage procedure aiming to minimize the posterior variance of the quantile estimator, unraveling into a difficult numerical problem requiring many approximations.

- Liu and Staum (2010) used stochastic kriging in estimating *tail value-at-risk* (TVaR_α) (which is simply a tail average) of a financial portfolio. They also use a three-stage procedure that minimizes the posterior variance of the TVaR_α estimator (requiring numerical approximations).

- Picheny et al. (2010) sought to understand the set $\{z : f(z) \in (L-\varepsilon, L+\varepsilon)\}$ where f is the unknown function and L is a known level. The similarity here is that we are interested in the case where L is VaR_α, so that we will gain inference about the z producing values of $f(z)$ in the neighborhood of L,

and can consequently pursue attention in this region. Unfortunately, L is not known in our example, and furthermore their method considers the case where $f(z)$ is non-noisy.

The last method is closely related to similar works of estimating an excursion set $\{z : f(z) \geq L\}$ (Chevalier et al. (2014a)) and estimating probability of failure (Bect et al. (2012)), i.e. $\mathbb{P}(\{z : f(z) \geq L\})$. Each incorporates a *stepwise uncertainty reduction* (SUR) procedure that sequentially pick a scenario z at which to evaluate $f(z)$, where z is chosen based on a specified expected improvement criterion.

All three methods use an initial step of using a fraction of the budget to learn about $f(z)$ on the whole domain, which yields an initial search region for the remainder of the procedure. The methods differ in allocating the remaining budget: Oakley (2004) performs one additional step that chooses design points for f to be evaluated at, based on what the initial stage inferred about the quantile region of f. Liu and Staum (2010) is similar in having two additional steps, where first a larger fraction of the budget is allocated uniformly to scenarios in the estimated tail, and finally the remaining budget is dispersed (non-uniformly) among the tail scenarios according to minimize the posterior variance of the TVaR_α estimator. As mentioned, Picheny et al. (2010) finishes by *sequentially* evaluating f at points chosen according to an expected improvement criterion.

In the VaR context, several aspects of the setting make the emulation problem statistically challenging. First, as mentioned the number of outer scenarios is quite large, rendering existing ranking-and-selection strategies from the OR literature computationally heavy. We note that the typical simulation budget is $N_{tot} \sim \mathcal{Z}$,

i.e. naively with a brute-force uniform approach one can only sample each scenario a handful of times, which will lead to disastrous estimates. This implies that practical allocation schemes must be quite *aggressive* and the asymptotic guarantees available in the above literature are not applicable. Second, the outer scenarios are usually treated by practitioners as a fixed object. In other words, the scenario space is taken to be discrete – one cannot add (or subtract) further scenarios (which is the strategy in Broadie et al. (2011)). Neither can one employ continuous search methods that are popular in the simulation optimization domain. Third, the nature of the underlying cashflows makes the simulation noise ϵ highly non-Gaussian. It is typically skewed (because many cashflows have embedded optionality and hence the corresponding distribution has a point mass at zero) and with low signal-to-noise ratio. The latter implies that cross-scenario information borrowing is crucial to maximize accuracy. Fourth, the portfolio losses are highly inhomogenous, i.e. ϵ is *heteroskedastic*. This requires an advanced emulator to overcome this challenge. Fifth, the statistical objective function which defines the emulator goodness-of-fit is highly non-standard. On the one hand, it is implicit because the goal is to find the critical threshold $\tilde{z}$ such that $\tilde{z} = f(z)_{(\alpha N)}$ On the other hand, there are no simple estimators for the empirical quantile, so uncertainty quantification for the $\tilde{z}$ is also nontrivial. These two points are crucial. The standard ranking-and-selection procedure is asymptotically equivalent to doing a hypothesis test for $f(z) > L$ for a given threshold L and for each scenario z. Theoretically this allows to decouple the estimation problems, but such schemes are obviously impractical if L is itself unknown. Consequently, an emulator-based, *sequential* procedure is instead needed.

In this article we propose a universal strategy that overcomes all of the above challenges. Our framework has 3 key pieces. First, we connect to the burgeoning DACE literature on level-set estimation, tailoring the recent successes of the Stepwise Uncertainty Reduction (SUR) techniques to the VaR/TVaR problem. As such, we see this work as a first step towards closer marriage of machine learning and simulation-based risk measurement. Second, as already mentioned we work with a GP emulator which offers a rich uncertainty quantification properties, in particular many analytic formulas for active learning criteria that are used for guiding the simulation allocation. Third, we *take advantage* of the discrete scenario set which intrinsically calls for a replicated design. In turn, replication yields (i) improved noise properties that minimize non-Gaussianity; (ii) ability to simultaneously learn the mean response $f(\cdot)$ and the conditional simulation variance $s(\cdot)$ to handle heteroskedasticity; (iii) reduced model overhead through possibility of batching; (iv) convenient GP implementation. The symbiotic relationship of replication and kriging was already observed in Ankenman et al. (2010); we further modify standard GPs by employing the `hetGP` library that was recently built to offer a tailored emulator that gracefully handles (i)-(iv).

Rather than proposing a single specialized algorithm, we present a general framework which contains several modules. By adjusting these "moving parts", the algorithm can (a) evaluate different risk-measures (we illustrate with both VaR and TVaR); (b) can use different emulator codes; (c) can change different ways of running the sequential budget allocation: initialization phase; number of sequential budget; termination criterion; batch-size and so on; (d) can rely on different VaR estimators. To illustrate above choices we present an extended

numerical illustration that compare the impact of different modules. In two case studies, we compare it with benchmarks such as the algorithm in Liu and Staum (2010). In sum, we will show that the emulation approach provides accurate estimates even with a small simulation budget, and that the methods used have a relatively low overhead numerical cost (fitting, prediction, etc.).

Returning to the applied context, our main message is that significant efficiency can be squeezed from the proposed *statistical tools*, and so nested simulation for risk measurement is a truly feasible paradigm. In turn, the statistical learning perspective implies a natural preference for *sequential* algorithms that repeatedly go through the feedback loop of: simulate – estimate – assess – simulate. Compared to classical one-stage/two-stage designs, the fully sequential strategies offer several attractive features. First, they internalize the uncertainty quantification that is offered by the emulator and therefore should (eventually) lead to more automated implementation. Second, they can be run in an online fashion – simulation run on a server or in the cloud with interim results always available for download and inspection while the algorithm proceeds to refine its accuracy. Thus, the user no longer has to specify a priori the simulation budget, but instead can interact with the code on-the-fly. Third, sequential methods can be seen as a starting point for more general re-use of simulations, for example for periodic recomputation of the whole problem as business time progresses (usually VaR calculations are done on a daily-weekly-monthly cycles).

The paper is organized as follows: in Section 5.2 we discuss the emulation objective and explain further the idea of portfolio loss and tail risk, along with VaR_α estimation. Section 5.3 formally introduces the GP model and its mathe-

matical details. Next, Section 5.4 summarizes the mathematical ideas in Liu and Staum (2010) and Picheny et al. (2010), along with how they are implemented and adapted to our problem. Section 5.5 provides the full implementation. Finally, we apply the algorithm to two case studies where VaR_α and $TVaR_\alpha$ calculations are of interest. Section 5.6 involves a two-factor model, where we compare methods for fixed simulation budgets, as well as a sequential stopping criteria. Section 5.7 investigates performance in higher dimensions where the state process is six-dimensional and the simulator is much more expensive.

5.2 Objective

Given a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, we consider a stochastic system with Markov state process $Z = (Z_t)$. Typically, Z is a multivariate stochastic process based on either a stochastic differential equation or time-series ARIMA framework. Based on the realizations of Z, the modeler needs to assess the available capital $-f(Z_T)$ at an intermediate horizon T. Noting that $f(Z_T)$ is the *loss* at time T, the *capital requirement* based on VaR_α and TVaR_α are respectively the value sufficient to cover time T liabilities with probability α, i.e.

$$\text{VaR}_\alpha(-f(Z_T)) = \inf\{x : \mathbb{P}(x + f(Z_T) \geq 0) \geq \alpha\}, \tag{5.2}$$

and the $\alpha-$tail average,

$$\text{TVaR}_\alpha(-f(Z_T)) = \mathbb{E}\left[-f(Z_T) | -f(Z_T) \leq \text{VaR}_\alpha(f(Z_T))\right]. \tag{5.3}$$

The portfolio value $f(Z_T)$ is computed as the conditional expected value of discounted cash flows given Z_T, the information at time T. For concreteness, consider cashflows $Y_t(z_\cdot)$ which could depend on the whole path $z_{0:t}$ of the stochastic factor up to date t. To compute their net present value, we discount at a constant risk-free rate β

$$Y(z) = \sum_{t=T}^{\infty} e^{-\beta(t-T)} Y_t^i(z_\cdot^n). \tag{5.4}$$

Since Z is Markov, we can write

$$f(z) \doteq \mathbb{E}\left[Y(Z_T)|Z_T = z\right]. \tag{5.5}$$

Aside from situations where the form of cashflows are trivial, we shall assume that $f(z)$ is not available explicitly, and there is no simple way to describe its functional form. However, since $f(z)$ is a conditional expectation, it can be sampled using a simulator, i.e. the modeler has access to an engine that can generate independent, identically distributed trajectories $(Y_t^n(z))_{t=T+1}^{\infty}$, $n = 1, 2, \ldots,$ given $Z_T = z$. However this simulator is assumed to be expensive, implying that computational efficiency is desired in using it.

The naive Monte Carlo approach is based on nested simulation. First, we start with N trajectories of $(Z_t)_{0\le t\le T}$, yielding $\mathbf{z} := \{z^n, n = 1, \ldots, N\}$. Next, for each n, $f(z^n)$ is approximated by an inner empirical average

$$f(z^n) \simeq \bar{y}^n \doteq \frac{1}{r^n} \sum_{i=1}^{r^n} y^{n,i}, \qquad n = 1, \ldots, N, \tag{5.6}$$

where $y^{n,i}$ is the present value of loss from (5.4) based on $i = 1, \ldots, r^n$ independent

trajectories $Y_{\cdot}^{i}(z_{\cdot}^{n,i})$. Finally, Equations (5.2) and (5.3) are estimated by a quantile estimator and tail average estimator respectively (e.g. the empirical αNth order statistic of the realizations $(\bar{y}^n)_{n=1}^N$ and average of the αNth lowest ordered values; more sophisticated estimators are discussed in Section 5.2.1).

Remark. Typically the outer scenarios are provided by an Economic Scenario Generator that calibrates the dynamics of (Z_t) to the historical data, i.e. it is under the physical measure $\mathbb{P}$. The inner simulations are based on a mark-to-market law, i.e. it is under the risk-neutral measure $\mathbb{Q}$. Therefore, the evolution of (Z_t) is different on $[0, T]$ and on $[T, \infty)$. This makes no difference for our subsequent discussion which takes the set (z^n) as given. Hence, the inner workings of the ESG/outer simulator are never considered.

Specifically for this nested approach, Gordy and Juneja (2010) provide results about asymptotic bias and variance, along with consequent optimal budgeting strategies, i.e. how to choose N and $r^n = \bar{r}, n = 1, \ldots, N$ for a fixed budget $N_{tot} \doteq \sum_{n=1}^{N} r^n$. Practically this approach is unreasonable expensive: with a uniform inner allocation the total simulation budget is $\mathcal{O}(N \cdot \bar{r})$ – for example, a budget of $\bar{r} = 10^3$ inner and $N = 10^3$ outer simulations requires 10^6 total simulations.

For this reason, it is desirable to construct more frugal schemes for approximating (5.2). The main idea is to replace the inner step of repeatedly evaluating $f(z)$ by a *surrogate model* $\hat{f}$ for f. This framework generates a fitted $\hat{f}$ by solving regression equations over a training dataset. Emulation reduces approximating f to the twin statistical problems of (i) experimental design (generating the training dataset) and (ii) regression (specifying the optimization problem that the approx-

imation $\hat{f}$ solves). The problem of statistical design is of particular importance to quantile estimation; the surrogate $\hat{f}$ should be accurate in "extreme" regions, so a procedure that identifies these scenarios and allocates budget accordingly is necessary. Details of these steps are presented in Sections 5.4 and 5.5 below. The model used for f is discussed in Section 5.3.

5.2.1 Tail Risk Estimation

Tail risk is the risk associated with extreme financial scenarios. In practice, corporations are regulated to hold sufficient capital in case of these events. In almost all cases, the capital requirement is calculated as a *risk measure* of future loss, see Dhaene et al. (2006) for a review. The exact risk measure depends on the industry and specific regulatory frameworks, but typically is based on VaR_α or TVaR_α.

The primary focus of this paper is to investigate estimation of VaR_α and TVaR_α, though our framework can straightforwardly handle generic risk measures of $\mathcal{Z}$ defined through weights: given a collection of losses (f^n) with $f^n = f(z^n)$ define

$$R \doteq \sum_{n=1}^{N} w^n f^n, \tag{5.7}$$

where $\sum_{n=1}^{N} w^n = 1$. For example, fixing α, the level of risk (and assuming αN is an integer), let $f^{(\alpha N)}$ be the αNth order statistic of $f^{1:N}$. Then VaR_α and TVaR_α

respectively have weights

$$w^{n,VaR} = \mathbb{1}_{\{n:f^n=f^{(\alpha N)}\}} \tag{5.8}$$

$$w^{n,TVaR} = \frac{1}{\alpha N}\mathbb{1}_{\{n:f^n\leq f^{(\alpha N)}\}}. \tag{5.9}$$

The empirical plug-in analogue of applying Equation (5.7) to the Monte Carlo estimates in Equation (5.6), i.e. using $\hat{f}_{(\alpha N)}$ produces a biased estimate, as discussed in Kim and Hardy (2007). The authors also discuss the difficulty that one cannot quantify which direction the bias lies without knowing some details about f and the distribution of Z_T. Kim and Hardy (2007) and Gordy and Juneja (2010) discuss ways of reducing bias, through bootstrap and jackknife, respectively.

Borrowing results from the general theory of order statistics, one can construct several modified versions of VaR_α estimators that stem from weighted averages of nearby order statistics. Called $L-$estimators Sheather and Marron (1990), they offer robustness compared to a single sample order statistic. Effectively such estimators modify the weights w^n defining (5.7) to account for the uncertainty in $\hat{f}(z^n)$. A well known construction is by Harrell and Davis (1982), where the weights for a VaR_α estimator are chosen as (note the link to the beta distribution)

$$w^{(n)} = \int_{n-1/N}^{n/N} \frac{\Gamma(N+1)}{\Gamma((N+1)\alpha N)\Gamma((N+1)(1-\alpha N))} t^{(N+1)\alpha N-1}(1-t)^{(N+1)(1-\alpha N)-1}dt, \tag{5.10}$$

where the index (n) is based on the order of the $\bar{y}^n, n = 1, \ldots, N$. Sheather and Marron (1990) provides details on the properties of the weights in Equation (5.10)

as well as a discussion on other $L-$estimators. Sfakianakis and Verginis (2008) provide more sophisticated $L-$estimators, as well as performance comparison with the Harrell-Davis estimators and the exact empirical estimator in (5.8). The general conclusion is that no one estimator performs best – they are all situational, with properties depending on the problem at hand.

In Section 5.3.5, we further discuss these risk measures in the case where the losses $Y^1, \ldots, Y^N$ are multivariate normal, which is the setup of Section 5.3.

5.3 Stochastic Kriging

The idea of emulation is to "learn" the response surface $z \mapsto \hat{f}(z)$ by a regression step that borrows information across different scenarios starting at various sites z^n. This reduces computational budget compared to the nested simulation step of independently making N pointwise estimates $f(z^n)$ by running r^n realizations from *each* site z^n. The result is a fitted surrogate $\hat{f}$ that smoothes the Monte Carlo noise from nearby scenarios. Moreover, the surrogate allows to forecast $\hat{f}(z)$ at scenarios where *no* inner simulations were used at all, thereby offering screening of scenarios that are far from the tail risk.

Formally, the statistical problem of emulation deals with a sampler (or oracle)

$$Y(z) = f(z) + \epsilon(z), \tag{5.11}$$

where we identify $f(z)$ with the unknown *response surface* and ϵ is the sampling noise, assumed to be independent and identically distributed across different calls to the oracle. We make the assumption $\epsilon(z) \sim N(0, \tau^2(z))$, where $\tau^2(z)$ is the

sampling variance that depends on the scenario z. Emulation now involves the (i) experimental design step of proposing a design $\mathcal{D}$ that forms the training dataset, and (ii) a learning procedure that uses the queried results $\mathcal{D}_k = (z^n, \mathbf{y}_k^n)_{n=1}^N$, with $\mathbf{y}_k^n = \{y_k^{n,1}, \ldots, y_k^{n,r_k^n}\}$ being a collection of r_k^n realizations of (5.11) given z^n, to construct a fitted response surface $\hat{f}$. Here, we consider the case where $\hat{f}$ is fitted sequentially based on a multi-step procedure, and subscript k denotes the step counter.

A kriging surrogate assumes that f in (5.11) has the form

$$f(z) = \mu(z) + X(z), \tag{5.12}$$

where $\mu : \mathbb{R}^d \to \mathbb{R}$ is a trend function, and X is a mean-zero square-integrable Gaussian process with covariance kernel C. The role of C is to generate the RKHS $\mathcal{H}_C$ which is the functional space that X is assumed to belong to.

Since the noise $\epsilon(z)$ is also Gaussian implies that $X(z)|\mathcal{D} \sim N(m(z), s^2(z))$ has a Gaussian posterior, which reduces to computing the kriging mean $m(z)$ and kriging variance $s^2(z)$. In turn, the kriging variance $s^2(z)$ offers a principled empirical estimate of model accuracy, quantifying the approximation quality. In particular, one can use $s^2(z)$ as the proxy for the MSE of $\hat{f}$ at z.

5.3.1 Predictive Distribution

By considering the process $f(z) - \mu(z)$, we may assume without loss of generality that f is statistically centered at zero. Denoting the sample average at each scenario z^n by $\overline{y}_k^n \doteq \frac{1}{r_k^n} \sum_{i=1}^{r_k^n} y_k^{n,i}$ as in Equation (5.6) and the overall collec-

tion as $\overline{\mathbf{y}}_k = \{\overline{y}_k^1, \ldots, \overline{y}_k^N\}$, the resulting posterior mean and variance of $\hat{f}_k(z)$ are Roustant et al. (2012a)

$$\begin{cases} m_k(z) \doteq \mathbf{c}(z)^T(\mathbf{C} + \mathbf{\Delta}_k)^{-1}\overline{\mathbf{y}}_k; \\ s_k^2(z) \doteq C(z,z) - \mathbf{c}(z)^T(\mathbf{C} + \mathbf{\Delta}_k)^{-1}\mathbf{c}(z), \end{cases} \tag{5.13}$$

where $\mathbf{c}(z) = (C(z, z^n))_{1 \le n \le N}$, $\mathbf{C} \doteq [C(z^i, z^j)]_{1 \le i,j \le N}$, and $\mathbf{\Delta}_k$ is the diagonal matrix with entries $\tau^2(z^1)/r_k^1, \ldots, \tau^2(z^N)/r_k^N$. Note that the conditional variances $\tau^2(\cdot)$ are typically unknown, so to obtain $s_k^2(z)$ it must be replaced with a further approximation $\hat{\tau}^2(\cdot)$ as discussed in Section 5.3.3.

5.3.2 Covariance kernels and hyperparameter estimation

The covariance function $C(\cdot, \cdot)$ is a crucial part of a Kriging model. In practice, one usually considers spatially stationary or isotropic kernels,

$$C(z, z') \equiv c(z - z') = \sigma^2 \prod_{j=1}^{d} g((z - z')_j; \theta_j),$$

reducing to the one-dimensional base kernel g. Below we use the Matern 5/2 kernel,

$$g(h; \theta) = \left(1 + \frac{\sqrt{5}h}{\theta} + \frac{5h^2}{3\theta^2}\right) \exp\left(-\frac{\sqrt{5}h}{\theta}\right). \tag{5.14}$$

The hyper-parameters θ_j are called characteristic length-scales and can be informally viewed as roughly the distance you move in the input space before the response function can change significantly (Rasmussen and Williams, 2006, Ch 2). Constructing a GP emulator requires picking a kernel family and the hyper-

parameters σ_j, θ_j. We utilize the `R` packages `DiceKriging` Roustant et al. (2012a) and `hetGP` Binois et al. (2016) that allow fitting of kriging models for several kernel families by Maximum Likelihood.

5.3.3 Intrinsic Variance

The simulation variance $\tau^2(z)$ is a crucial piece of the emulation but is again unknown. A basic estimate can be obtained as the empirical standard deviation across the r_k^n inner simulations:

$$\hat{\tau}_k^2(z^n) = \frac{1}{r_k^n - 1} \sum_{i=1}^{r_k^n} (y_k^{n,i} - \bar{y}_k^n)^2, \tag{5.15}$$

so that our proxy for the intrinsic variance of $\hat{f}_k(z^n)$ is $\hat{\tau}_k^2(z^n)/r_k^n$. Depending on the circumstances, this sample variance estimator is unreliable, in particular, when r_k^n is small. Since the noise $\tau^2(z)$ should be similar at nearby scenarios, a hierarchical approach can be used where $\tau^2(z)$ itself is modeled by a kriging surrogate. Brief details are given in Section 3.1 of the seminal work of Ankenman et al. (2010) on stochastic kriging. More recently, Binois et al. (2016) looks at this problem in detail and provides efficient numerical formulas to an otherwise expensive addition to the stochastic kriging framework.

5.3.4 Updating Equations

In the setting where $\hat{\tau}^2(\cdot)$ itself is not modeled, it is numerically more efficient to keep track of $\bar{y}^n$ and $\hat{\tau}^2(z^n)$ instead of all replications $\mathbf{y}^n$ at z^n. Let $r_k'^n$ be the number of replications to be added to scenario z^n. Then, for the sites with

$r'^n_k > 0$, let $\bar{y}'^n_k$ and $\hat{\tau}'^2_k(z^n)$ be the sample mean and standard deviation of the new replications. Then one can update with the following equations (Chan et al. (1982))

$$\bar{y}^n_{k+1} = \frac{r^n_k \bar{y}^n_k + r'^n_k \bar{y}'^n_k}{r^n_k + r'^n_k} \tag{5.16}$$

$$\hat{\tau}^2_{k+1}(z^n) = \frac{1}{(r^n_k + r'^n_k - 1)} \left((r^n_k - 1)\hat{\tau}^2(z^n_k) + (r'^n_k - 1)\hat{\tau}'^2_k(z^n) + \frac{r'^n_k r^n_k}{r^n_k + r'^n_k} (\bar{y}^n_k - \bar{y}'^n_k)^2 \right). \tag{5.17}$$

Equation (5.17) also showcases how each individual piece contributes to the noise variance. In this case, one must also keep track of r^n_k.

The Gaussian process itself also has updating equations for the new posterior mean and variance when a new observation is added to the design, see Kamiński (2015) for details.

5.3.5 Tail Risk Estimation in the Gaussian Process Case

Returning to the risk measure as defined in (5.7), where $w^n = c\mathbb{1}_{\{z^n \in \mathfrak{R}\}}$ its posterior is

$$\mathbb{E}[R|\mathcal{D}_k] = \sum_n \mathbb{E}[cf(z^n)\mathbb{1}_{z^n \in \mathfrak{R}}|\mathcal{D}_k] \tag{5.18}$$

which requires integrating against the joint distribution of $\hat{f}^{1:N}$. While the latter is multivariate Gaussian, there are no closed formulas for the probability that one coordinate of a MVN distribution is a particular order statistic. Nevertheless, the conditional expectation can be numerically approximated by making draws

from the above posterior MVN of $\hat{f}^{1:N}$, evaluating the resulting quantile/tail and averaging. A slightly cheaper procedure is to treat the two terms in (5.18) as independent:

$$\begin{aligned}\mathbb{E}[R|\mathcal{D}_k] &\simeq \sum_n \mathbb{E}[cf(z^n)|\mathcal{D}_k] \cdot \mathbb{E}[\mathbb{1}_{z^n \in \mathfrak{R}}|\mathcal{D}_k] \\ &= cm_k(z^n) \cdot \mathbb{P}(z^n \in \mathfrak{R}|\mathcal{D}_k) \doteq c\omega_k^n m_k(z^n), \end{aligned} \tag{5.19}$$

where the weights $\omega_k^n = \mathbb{P}(z^n \in \mathfrak{R}|\mathcal{D}_k)$ resemble the Harrell-Davis construction. As the emulator learns f, we expect that $\omega_k^n \to w^n$ converge to the true weights. Figure 5.1 in Section 5.6 provides a visual comparison of $\omega_k^n, \tilde{w}_k^n$ and $W_k(z^n)$, a weight defined in Section 5.4.1. In general, it shows that the Harrell-Davis weights provide a shape much similar to the true ω_k^n.

To reduce numerical costs, we use a hybrid method substituting the Harrell-Davis weights in Equation (5.10) for $\omega_k^n = \tilde{w}_k^n$ which are in turn based on the order statistics of the posterior means $m_k(z^n)$. The figure shows that these are in fact quite close and offers a more robust quantile estimator, henceforth labeled $\hat{R}_k^{HD}$. For further self-assessment we also compute the estimator variance:

$$s_k^2(R) \doteq \text{var}(R|\mathcal{D}_k) \doteq \mathbf{w}_k \left[\mathbf{C} - \mathbf{c}(z)^T(\mathbf{C} + \mathbf{\Delta}_k)^{-1}\mathbf{c}(z)\right] \mathbf{w}_k^T, \tag{5.20}$$

where $\mathbf{w}_k = (w_k^1, \ldots, w_k^N)$. Note that the inside term in Equation (5.20) is the posterior covariance matrix of $\hat{f}_k(\mathbf{z})$, taking advantage of the covariance structure of the GP for additional smoothing. For estimating TVaR_α, the empirical weights defined through Equation (5.9) are used. Theoretically, we could utilize weights that trail off near $m_k^{(\alpha N)}$ to create a structure similar to Harrell-Davis, however,

the empirical TVaR_α estimator defined in Equation (5.9) is already robust in that misspecification of a few boundary scenarios is minor compared to incorrectly choosing the single index for VaR_α.

Finally below we also discuss the quantile scenario, i.e. the $z \in \mathcal{Z}$ corresponding to αN order statistic. We denote by $\tilde{z}$ and $\hat{z}_k$ the *true* αN *tail scenario* (based on $f^{1:N}$) and *estimated* αN *tail scenario based on* $m_k(z^{1:N})$ respectively.

5.4 Sequential Design for Tail Approximation

Let N_{tot} be the total budget which is split into K sequential rounds $k = 1, \ldots, K$. We work with a fixed set of outer scenarios $\mathcal{Z} \doteq (z^1, \ldots, z^N)$. An outline of the general procedure proposed to estimate $\hat{R}_K$ (as in Equation (5.19)) is given as follows:

1. Initialize $\hat{f}_1$ by simulating some simulations for a subset of *pilot* scenarios.

2. Sequentially over $k = 1, 2, \ldots$ until the N_{tot} budget is depleted, predict $\hat{f}_k$ on $\mathcal{Z}$ to determine which scenarios might be close to $\tilde{z}$. Allocate more inner simulations to these scenarios, i.e. increase the corresponding r^n's. This will be achieved via an *acquisition function* that takes into account the "closeness" of z^n to $\tilde{z}$ and the uncertainty $s_k(z^n)$. Potentially a new outer scenario (that previously had $r_k^n = 0$) might be searched. Then update to produce $\hat{f}_{k+1}$ based on the new data.

3. The final estimate is obtained from Equation (5.19), with uncertainty expressed via (5.20).

Making the above mathematically precise boils down to two key objectives: 1. discover the region of $\mathcal{Z}$ where $\tilde{z}$ lies (or the tail in case of TVaR_α), and 2. reduce $s_k^2(\cdot)$ in this region. These two objectives match the exploration-exploitation tradeoff: allocating too many replications to solve (1) produces a surrogate that lacks precision even though it recognizes the location of $\tilde{z}$, while focusing only on (2) without sufficient searching may focus on the wrong region. To guide this trade-off during sequential allocation, the acquisition function is based on an uncertainty measure. The strategy is then to (myopically) carry out *stepwise uncertainty reduction* (SUR) by determining what new simulations would most reduce expected uncertainty for the *next* round. See Bect et al. (2012); Chen et al. (2012); Chevalier et al. (2014a); Liu and Staum (2010); Oakley (2004); Picheny et al. (2010).

Some strategies discussed below require evaluating a quantity dependent on the posterior covariance matrix in Equation (5.13) repeatedly for various $z \in \mathcal{Z}$ to see which provides the best improvement; in this case, repeating the matrix inversion at each step is expensive computationally, but the updating Equations in Section 5.3.4 provide efficient calculations.

For computational efficiency, rather than adding a single inner simulation, we work with batches of size Δr. For example, $\Delta r = 0.01 N_{tot}$ yields a procedure with $K = 100$ rounds. One approach is to sequentially pick $z \in \mathcal{Z}$ and allocate all Δr new replications to that scenario. Although the updating equations are numerically efficient, the majority of the methods below introduce a non-negligible computational expense, so that calculating the optimal selection for each replication is unfeasible.

5.4.1 Stepwise Uncertainty Reduction for Risk Measures

Define

$$V_k(z^m|z^n) \doteq C(z^m, z^m) - \mathbf{c}(z^m)^T(\mathbf{C}+\mathbf{\Delta})^{-1}\mathbf{c}(z^m)\Big|_{\mathbf{\Delta}=\text{diag}\left(\frac{\hat{\tau}_k^2(z^1)}{r_k^1},\ldots,\frac{\hat{\tau}_k^2(z^n)}{r_k^n+\Delta r_k},\ldots,\frac{\hat{\tau}_k^2(z^N)}{r_k^N}\right)} \tag{5.21}$$

which approximates the kriging variance at z^m under the assumption that Δr_k replications were added to z^n (keeping all other GP pieces frozen from the k-th round).

Our goal is to learn R in (5.7). A key challenge is that R is defined via the set $\mathfrak{R}$ which in turn depends on all of $f(z^{1:N})$. In existing literature, a more common criterion is related to a fixed threshold L with $\Gamma_L \doteq \{z : f(z) \geq L\}$, and the search done over a continuous domain with deterministic simulations. Depending on the perspective, one may estimate (i) the *contour* $\partial\Gamma_L = \{z : f(z) = L\}$ Bect et al. (2012); (ii) the excursion volume $\mathbb{P}(\Gamma_L)$ Chevalier et al. (2014a); (iii) the level or excursion set Γ_L Picheny et al. (2010). A comprehensive implementation of all above objectives can be found in Chevalier et al. (2014b). More recently, Labopin-Richard and Picheny (2016) produced a work where the interest is on finding the level itself in a non-noisy setting. They use two criteria, the main one being a slight modification based on the criteria developed in Bect et al. (2012). The other is not applicable here since it depends specifically on being in a non-noisy setting.

For the VaR objective, $\mathfrak{R}$ is essentially a contour; for TVaR it is essentially the level set. As we show below, the acquisition functions corresponding to (i)-(ii) yield similar performance. Chevalier et al. (2014b) states that because of the difficulties to compute the criteria for (iii), it offers roughly the same performance

as for (ii). Due to its expense, we do not implement it in this paper. We focus on the contour problem since it is most related to ours and also allows for uncertainty in the level L; the level set and volume uncertainties and criteria are provided in 5.9.

Remark. The above papers are interested in the case where $\mathcal{Z}$ is continuous, so the designs augment with new additional sites z^{k+1} in $\mathcal{Z}$. In our case $\mathcal{Z}$ is finite, so we add *replications* to existing $z \in \mathcal{Z}$. Consequently, integrals are replaced by sums over $\mathcal{Z}$.

Our main criterion is based on the *targeted mean square error* at a scenario z:

$$\begin{aligned} \mathrm{tmse}_k(z) :&= s_k^2(z) \frac{1}{\sqrt{2\pi(s_k^2(z)+\varepsilon^2)}} \exp\left(-\frac{1}{2}\left(\frac{m_k(z)-L}{\sqrt{s_k^2(z)+\varepsilon^2}}\right)^2\right) \\ &= s_k^2(z) W_k(z;L). \end{aligned} \tag{5.22}$$

The parameter ε controls how localized is the criterion around the level L. Note that $\varepsilon = 0$ is very local. The numerical examples of Picheny et al. (2010) choose ε to be five percent of the output range, however, they consider the case $\tau^2 \equiv 0$. In our case, it is desirable to have ε decrease as k increases, since knowledge about the level and its surrounding region improves. We propose to take $\varepsilon = s_k(R)$ which captures the uncertainty about the quantile. This is large when k is small (uncertain in earlier stages), and decreases rapidly as k increases (gaining certainty about L).

Observe that $W_k(z;L) = \phi(m_k(z)-L, s_k^2(z)+\varepsilon^2)$ is based on the Gaussian pdf ϕ, and is largest for scenarios where portfolio value is close to L and scenarios that

have higher posterior variance. The goal is now to reduce the kriging variance, but do it for scenarios close to L. This strategy was originally introduced in Picheny et al. (2010) for estimating the contour set Γ_L. Specifically, the acquisition function to be minimized is $\mathcal{H}_{k+1}$ where

$$\mathcal{H}_{k+1}^{\text{timse}} := \frac{1}{N}\sum_{n=1}^{N} \text{tmse}(z^n) = \frac{1}{N}\sum_{n=1}^{N} s_{k+1}^2(z^n) W_{k+1}(z^n; R), \tag{5.23}$$

i.e. the average $\text{tmse}(z)$ over all $z \in \mathcal{Z}$. Practically, we rely on the approximate predictive variance in (5.21), the current timse weight $W_k(z^n)$ and the plug-in VaR estimator $\hat{R}_k$ to minimize

$$\text{timse}_k(z) \doteq \frac{1}{N}\sum_{n=1}^{N} V_k(z^n; z) W_k(z^n; \hat{R}_k). \tag{5.24}$$

For TVaR all scenarios in the tail $z^n \in \Gamma$ need to be considered so we modify the criterion in Equation (5.24) to instead use weights

$$\begin{aligned} \text{tmse}_k^{\text{TVaR}}(z) :&= s_k^2(z) \frac{1}{\sqrt{2\pi(s_k^2(z)+\varepsilon^2)}} \left(1 - \Phi\left(\frac{m_k(z) - L}{\sqrt{s_k^2(z)+\varepsilon^2}}\right)\right) \\ &\simeq V_k(z^n; z) W_k^{\text{TVaR}}(z; \hat{R}_k), \end{aligned} \tag{5.25}$$

where $\Phi(\cdot)$ denotes the standard normal cdf. Here, the weight function increases as z becomes deeper in the tail (i.e. $m_k(z) - L$ is more negative), while still compensating for uncertainty.

We remark that the criteria used for estimating the level set $\Gamma_L \doteq \{z : f(z) \geq L\}$ is inappropriate here, since interest is only in whether $f(z)$ lies on one side or the other of L. In other words, it neglects scenarios satisfying $f(z) \gg L$ because

they are so far beyond the level, even if they have reasonably high variance. Consequently, a TVaR estimator, which desires accuracy at all scenarios in the level set, would suffer.

5.4.2 Dynamic Allocation Designs

Batching allows the possibility of adding inner simulations in parallel for several different scenarios. To do this, let Δr_k be the budget for step k in the sequential procedure, and choose $\{r_k'^n\}_{n=1}^N$ that maximize an improvement criterion that depends on *all* $\{r_k'^n\}_{n=1}^N$ subject to the constraints $\sum_{n=1}^N r_k'^n = \Delta r_k$, and $r_k'^n \geq 0$ for all $n = 1, \ldots, N$. We define the improvement criteria as minimization of the posterior estimator variance $s^2(R_{k+1})$ in Equation (5.20). Note that $\boldsymbol{\Delta}_{k+1} = \mathcal{T}_{k+1}\boldsymbol{I}\underline{\mathbf{r}}_{k+1}^T$, where $\mathcal{T}_{k+1} = \left(\hat{\tau}_{k+1}^2(z^1), \ldots, \hat{\tau}_{k+1}^2(z^N)\right)$, $\boldsymbol{I}$ is the $N \times N$ identity matrix, and $\underline{\mathbf{r}}_{k+1} = \left(\frac{1}{r_k^1 + r_k'^1}, \ldots, \frac{1}{r_k^N + r_k'^N}\right)$, so the arguments to minimize appear only in $\underline{\mathbf{r}}_k$. This is a difficult optimization problem due to the matrix inversion and N being large; however, following Liu and Staum (2010) we provide an approximation that yields a closed form solution, see 5.10 for details. This approximation becomes more accurate as each r_k^n, $n = 1, \ldots, N$ increases, meaning it improves as the sequential procedure progresses.

By Lemma 5 in 5.10, the optimization reduces to minimizing

$$\mathbf{w}_{k+1}^T(\mathbf{C} + \mathcal{T}_{k+1}\boldsymbol{I}(\underline{\mathbf{r}}_k)^T)^{-1}\mathcal{T}_{k+1}\underline{\mathbf{r}}_{k+1}(\mathbf{C} + \mathcal{T}_{k+1}\boldsymbol{I}(\underline{\mathbf{r}}_k)^T)^{-1}\mathbf{w}_{k+1} \tag{5.26}$$

with respect to $r_k'^1, \ldots, r_k'^N$, given constraints $\sum_{n=1}^N r_k'^n = \Delta r_k$ and $r_k'^n \geq 0$ for $n = 1, \ldots, N$.Note that the $r_k'^n$ appear only in the middlemost term $\underline{\mathbf{r}}_k$. The

solution to this minimization problem is provided as Algorithm 3 in 5.10.

5.5 Algorithm

At each step, the procedures mentioned in Section 5.4 specify the number of replications $(r_k'^n)_{n=1}^N$ to add to each scenario. This choice actually summarizes the procedure completely, since we take the set of scenarios considered in step k as $\{z^n : r_k'^n > 0\}$; note that for the single point procedures there is only a single z^n such that $r_k'^n > 0$. At each step, the next set of values are

$$r_{k+1}^n \leftarrow r_k^n + \Delta r_k^n, \tag{5.27}$$

$$\mathbf{y}_{k+1}^n = \mathbf{y}_k^n \cup \{y_{k+1}^{n,r_k^n+1}, \ldots, y_{k+1}^{n,r_{k+1}^n}\}. \tag{5.28}$$

Then $\hat{f}_{k+1}$ is fit using the values

$$\bar{y}_{k+1}^n = \frac{1}{r_{k+1}^n} \sum_{i=1}^{r_{k+1}^n} y_{k+1}^{n,i}, \tag{5.29}$$

$$\hat{\tau}^2(z_{k+1}^n) = \frac{1}{r_{k+1}^n - 1} \sum_{i=1}^{r_{k+1}^n} (y_{k+1}^{n,i} - \bar{y}_{k+1}^n)^2. \tag{5.30}$$

in Equations (5.13). Algorithm 1 illustrates the resulting loop of adding more inner simulations, updating the fit $\hat{f}_k$ and then the acquisition function $\mathcal{H}_k$.

Thus, the general procedure is to repeatedly apply Algorithm 1, first for an initialization stage, and then using sequentially one of the methods described in Section 5.4, until the budget is depleted.

Algorithm 1: Algorithm to update $\mathcal{D}_k$

input : N_{tot}, $\hat{f}_k$, $\mathcal{D}_k$, $\{r_k'^n\}_{n=1}^N$
output: N_{tot}, $\hat{f}_{k+1}$, $\mathcal{D}_{k+1}$

for $n \in \{n : r_k'^n > 0\}$ **do**
  for $i = 1$ **to** $r_{k+1}'^n$ **do**
    Generate $(C_t^i(z^n))_{t=T}^{\infty}$;
    Compute $y_k^{n,i}$ via Equation (5.4);
  end
  Update $\mathbf{y}_{k+1}^n$ and r_{k+1}^n via Equations (5.28) and (5.27);
  $N_{tot} \leftarrow N_{tot} - \mathbf{\Delta} r_k^n$;
end
$\mathcal{D}_{k+1} \leftarrow (z^n, \mathbf{y}_{k+1}^n)_{n=1}^N$;
Update and return $\hat{f}_{k+1}$;

5.5.1 Initialization of $\hat{f}$

Initially, we have $r_0^n = 0$ for all n, i.e. no data to inform us where the tail region may lie. To initialize Algorithm 1, the typical solution is to use a small number of *pilot* scenarios that are representative of the entire domain $\mathcal{Z}$, determined by some space-filling algorithm (e.g. Latin Hypercube sampling or LHS). Chauvigny et al. (2011) provides a more ambitious method using statistical *depth functions* to identify pilot scenarios based on the geometry of $\mathcal{Z}$.

We found good performance from a simple space-filling approach. A small challenge is that LHS or even uniform sampling is not directly applicable with a discrete scenario place. Instead we develop a minimax-style procedure based on Euclidean distance described in Algorithm 2, where N_1 is the number of stage 1 scenarios, so that $\Delta r_1/N_1$ is the number of replications to add to each stage 1

scenario. We also let $\mathbf{z} = (z^1, \ldots, z^N)$, and standardize it to obtain

$$\mathbf{z}_{std} = \Sigma_z^{-1/2}(\mathbf{z} - \mu_{\mathbf{z}}), \tag{5.31}$$

where Σ_z and $\mu_{\mathbf{z}}$ are the sample covariance matrix and sample mean of $\mathbf{z}$. This allows $d(\cdot, \cdot)$, Euclidean distance between two points in $\mathbb{R}^n$, to make sense. Here, d_0 is a distance threshold that newly added points must fulfill from all other points included in the design. If d_0 is too small, then the algorithm may not search far enough, and it if it is too large, then the algorithm may not converge.

Algorithm 2: Determining stage 1 design using space-filling

input : $\mathbf{z}$, Δr_1, N_1, N_{tot}, d_0
output: N_{tot}, $\mathcal{D}_1$, $\hat{f}_1$

Uniformly randomize the order of $\mathbf{z}$;
Compute $\mathbf{z}_{std}$;
$I \leftarrow \{1\}$;
$j \leftarrow 2$;
while $|I| < N_1$ **do**
  if $\prod_{i \in I} \mathbb{1}_{\{d(z^j_{std}, z^i_{std}) \geq d_0\}} = 1$ **then**
    $I \leftarrow I \cup \{j\}$;
  end
  $j \leftarrow j + 1$;
end
Call Algorithm 1 with $\mathcal{D} = (z^n, \{\})_{n=1}^N$, $r'^n \equiv \Delta r_1 / N_1, n \in I, r'^n \equiv 0, n \notin I$;
Receive $N_{tot}, \mathcal{D}_1, \hat{f}_1$

5.5.2 Implementation Details

The classical method for inferring the hyperparameters $\boldsymbol{\theta}, \sigma^2$ is by optimizing the marginal likelihood, either through MLE or penalized MLE, using the likeli-

hood function based on the distributions described in Section 5.3.1. Either case leads to a nonlinear optimization problem to fit θ_j and process variance σ^2. One can also consider Bayesian Kriging, where trend and/or covariance parameters have a prior distribution, see Helbert et al. (2009). We utilize the both the `R` package "DiceKriging" Roustant et al. (2012a) as well as "hetGP" Binois et al. (2016). In a sequential setting, the hyperparameters are ideally refitted at each step, using new estimates if they yield a better likelihood function value, however this is computationally impractical since evaluation of the log-likelihood requires $\mathcal{O}(|\mathcal{D}_k|^3)$, where $|\mathcal{D}_k|$ is the size of the design set $\mathcal{D}_k$ used to fit the GP at step k. Alternatives to this are to refit the hyperparameters at certain points in the experiment, e.g. after 10%, 20%, ..., 90% of the budget has been depleted. An alternative is to use a nonlinear schedule, such as after 2%, 4%, 8%, ... of the budget has been depleted. This has the advantage of refitting more more frequently earlier when there is less certainty in the hyperparameter estimates.

Practitioners typically use the Matern 5/2 kernel in Equation (5.14) or the Gaussian kernel,

$$g(h, \theta) = \exp\left(\frac{-h^2}{2\theta^2}\right). \tag{5.32}$$

These kernels affect properties such as smoothness of sample paths, so this could be one criteria for choosing a kernel if there is prior knowledge about the true function f. For example, the Gaussian kernel is infinitely differentiable (in mean-square), while the Matern$-\nu$ kernel is differentiable at order k if and only if $\nu > k$ (so our choice is 5/2 times differentiable). In most cases, the choice of covariance kernel has only a minor impact on the resulting model, mostly due to the fact that the hyperparameter estimates will scale accordingly to whatever

spatial dependence and fluctuations the data represents.

One can choose a mean function through prior knowledge about f, or through data visualization. A parametric mean function can also be specified via basis functions with hyperparameters to be fitted, so that $\mu(z) = \beta_0 + \sum_{j=1}^{p} \beta_j h_j(z)$. In this case, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is estimated simultaneously with the other hyperparameters. In general, spatial closeness tends to dominate the posterior mean, so that the mean function has little impact in prediction aside from extrapolation. Still, a reasonably accurate mean function is desired for accurate hyperparameter fitting, since the θ parameters influence spatial dependence, something that detrending would affect.

One efficiency trick is to reduce $\mathcal{Z}$ to a candidate set $\mathcal{Z}^{cand}$ when evaluating the acquisition function. Note that computing the latter requires computing the predictive mean/covariance of the GP $\hat{f}_k$ which requires $\mathcal{O}(N_{fit}^2|\mathcal{Z}^{cand}|)$ work. One way to perform this screening is to compute weights, for example in Equation (5.22) (or (5.25) for TVaR) and only consider those that produce weights beyond a specified threshold, say 10^{-10}. This number can be modified depending on desired efficiency; however, we find numerically that 10^{-10} works well and reduces the candidate set by more than a factor of 10. Note that since the weights for VaR are not defined as probabilities, they should first be normalized and then compared to the threshold.

5.5.3 Comparison with other Approaches

To benchmark the proposed algorithms based on the SUR criteria defined in Section 5.4, we compare to several alternatives. We primarily focus on other

regression-based approaches and concentrate on quantifying the role of (i) adaptive budget allocation; (ii) sequential approaches as compared to simpler 1-, 2- or 3-stage methods. As an upper bound on performance, for VaR_α we include the Monte Carlo estimator under perfect information, i.e. perfect foresight regarding $\tilde{z}$, which therefore allocates the entire budget to a single scenario yielding $R_{best} \doteq \frac{1}{N_{tot}} \sum_n y^n$.

1. U1-GP: A 1-stage Uniform sampler. This method allocates $N_{tot}/|\mathcal{Z}|$ to each $z \in \mathcal{Z}$, fits a kriging model $\hat{f}_1$ to the resulting design and returns $m_1^{(\alpha N)}$. Comparing to this approach quantifies the gain of multi-stage procedures.

2. U2-GP: A 2-stage approach. The first stage is done as in Section 5.5.1. The pilot simulations are then used to screen scenarios for the main Stage-2 which allocates the entire remaining budget uniformly among the top $2\alpha N$ points. Thus, $r_2^n = N' 1_{\{m_1(z^n) > m_1^{(2\alpha N)}\}}$. This is the simplest version of an adaptive allocation.

3. A3-GP: The adaptive 3-stage algorithm of Liu and Staum (2010). The first stage uses space-filling pilot scenarios like in U2-GP; the second stage uniformly allocates inner simulations among a candidate set. The third stage then solves the global allocation problem defined in Appendix 5.10 to minimize variance of R_3. This method was designed for TVaR_α, but is easily adapted to VaR_α.

4. SR-GP: A sequential, rank-based algorithm. The allocation of new scenarios is based on the posterior means m_k. Specifically, for given ranks $L \leq \alpha N \leq U$, the algorithm allocates uniformly to all scenarios with

$\{z^n \in \mathcal{Z} : m_k(z^n) \in [m_k^{(L)}, m_k^{(U)}]\}$. Different values of L, U represent the particular version of SR-GP. For instance, taking $L = U = \alpha N$ gives a very aggressive scheme for estimating VaR: it greedily adds all Δr_k scenarios to the empirical quantile, i.e. scenario $\hat{z}_k$. (Note that U2-GP can be seen as a version of SR-GP with $K = 2$ rounds and $L = 1, U = 2\alpha N$)

5. True-SA: a perfect information estimator which knows the tail/quantile scenarios and sequentially works to minimize MSE of $\hat{R}$. For the VaR this corresponds to minimizing the posterior variance at $\tilde{z}$ which is trivially achieved by allocating the entire budget N_{tot} to the true quantile scenario. This is treated as the best possible algorithm which only has the averaging error due to the stochastic noise in $Y(z)$. For TVaR, the best allocation is roughly to spend equal budget on each tail scenario, modulo the spatial structure of $\mathcal{Z}$ (which makes estimating spatially-outlying scenarios harder).

A summary of these benchmarks, as well as the procedures discussed previously in the section are given in Table 5.1, with the appropriate parameters and explanations in Table 5.2.

The jn method in 5.9 was not included since the numerical overhead for its improvement criteria was too high; initial experiments showed it took longer than the next most expensive method by a factor of 8.5. Furthermore, Chevalier et al. (2014b) states that because of the difficulties to compute the jn criteria, it offers roughly the same performance as the sur criterion.

We also consider two non GP methods, one being the same as U1-GP but using the sample means $\bar{y}^n$ in place of a GP fit with prediction, this is called U1-SA. The second is a well known sequential algorithm in risk management, introduced

Name	Description
ST-GP	timse reducing (Equation (5.24))
SE-GP	sur reducing (5.9)
SV-GP	variance reducing (Section 5.4.2)
U1-GP	Naive Monte Carlo
SR-GP-1	Allocates only to $\hat{z}_k$ each time
U2-GP	Two-stage
SR-GP-2	Conservative sequential benchmark
A3-GP	Liu and Staum (2010)
True-GP	Monte Carlo under perfect information

Table 5.1: Summarizing the GP algorithms used as described in Section 5.5.3. Each sequential approach uses the same initialization procedure described in Algorithm 2 with $0.01N$ stage 1 scenarios. The budgeting parameters for the sequential methods are chosen to yield 100 total stages after the first stage (resulting in $K = 101$).

Name	Parameters
ST-GP	t $\varepsilon_k = s(R_k)$
U1-GP	Uses kriging for final prediction
SR-GP-1	L=U=50 (VaR), L=1,U=50 (TVaR)
U2-GP	Uniform allocation among top 100 ordered $m_1(z^n)$
SR-GP-2	L=26, U=75 (L=1, U=100) ordered $m_k(z^n)$ for VaR (TVaR)
A3-GP	70% of budget for stage 3; $n_0 = 100$

Table 5.2: Parameters and explanations (when applicable) for the algorithms in Table 5.1).

in Broadie et al. (2011). With our notation, the algorithm simply assigns $r'^n_k = 1$ to $n = \text{argmin}_n\, r^n_k|\bar{y}^n_k - L|/\tau(z^n)$ for round k, and $r'^n_k = 0$ otherwise. Here L is the level; they assume it is known, so we use the Harrell-Davis estimator in its place. They give suggestion to using a weighted estimate for $\tau(z^n)$ based on the aggregate variances gathered, but we use the sample variance in Equation (5.15) since our case studies imply a heteroskedastic variance surface. This algorithm is referred to as BR-SA. Comparison of GP methods to these are discussed in 5.6.1

5.6 Case Study: Black Scholes Option Portfolio

We begin the case studies with a two–dimensional example where $f(z)$ can be computed exactly, yielding a true comparative benchmark. We consider a portfolio whose value is driven by two risky assets that have Geometric Brownian motion dynamics:

$$dS_t^1 = \beta S_t^1\, dt + \sigma_1 dW_t^{(1)},$$
$$dS_t^2 = \beta S_t^2\, dt + \sigma_2 dW_t^{(2)}.$$

Above the W^i are correlated Brownian motions under the risk neutral measure with $d\langle W^{(1)}, W^{(2)}\rangle_t = \rho dt$. Other model parameters are summarized in Table 5.3. The portfolio consist of Call options: long 100 $K^1 = 85-$strike Calls on S^1 and short 50 $K^2 = 85-$strike Calls on S^2. We work with a risk horizon $T = 1$ year, and VaR_α and TVaR_α risk measures with $\alpha = 0.005$. By risk-neutral pricing, the value of the portfolio at T is

$$\Pi(z) \doteq \mathbb{E}^{\mathbb{Q}}\left[e^{-\beta(T_1-T)}\, 100\left(S_{T_1}^1 - 40\right)_+ - e^{-\beta(T_2-T)} 50\left(S_{T_2}^2 - 85\right)_+ \middle| (S_T^1, S_T^2) = z\right], \tag{5.33}$$

where $z \equiv (z^1, z^2) \in \mathbb{R}_+^2 =: \mathcal{Z}$. $\Pi(z)$ can be evaluated exactly using the Black Scholes formula; for our purposes Monte Carlo estimates can be obtained by simulating the log-normal values of S_2^1, S_3^2 conditional on $(S_1^1, S_1^2) = (z^1, z^2)$.

Asset	Position	Initial Price S_0^i	Strike K^i	Maturity T^i	Volatility σ^i
S^1	100	50	40	2	25%
S^2	-50	80	85	3	35%
	Correlation $\rho = 0.3$		Int. Rate $\beta = 0.04$		

Table 5.3: Parameters of the 2-D Case Study for a Black-Scholes portfolio on stocks S^1 and S^2.

5.6.1 Method & Results

Letting $f(z) = \Pi(z)$ we run all the methods listed in Table 5.1. The global parameters are $N = 10000$, $N_{tot} = 10^4$, so that $\alpha N = 50$. For $\mathcal{Z}$ we generate a *fixed* sample from the bivariate log-normal distribution of (S_1^1, S_1^2) under $\mathbb{Q}$ which is then re-used across all the methods. Note that the above implies that the dynamics of the factors on $[0, T]$ and $[T, T_i]$ are the same, i.e. the physical and risk-neutral measures coincide. This is solely for a simpler presentation of the case-study; in our experience the role of $\mathcal{Z}$ is secondary to the other considerations.

The Black-Scholes setup yields a closed-form solution for Equation (5.33), so we have a direct formula for the true quantile $\tilde{z}$ and corresponding $R_{BS}^{\text{VaR}} \doteq f^{(50)}$ and $R_{BS}^{\text{TVaR}} \doteq \frac{1}{50}\sum_{n=1}^{50} f^{(n)}$, as the exact 0.005 percentile of loss and 50 point tail average given $\mathcal{Z}$. Thus the bias and squared error over a single run can be computed exactly:

$$\text{bias}(R_k) = m(R_k) - R_{BS}, \tag{5.34}$$

$$\text{SE}(R_k) = \left(m(R_k) - R_{BS}\right)^2. \tag{5.35}$$

To stabilize results and reduce computation time, we assume the GP hyperparameters are known throughout all experiments. In practice, one does not know

these values ahead of time and they need to be re-estimated at each step, see the comments in Section 5.5.2. To determine the fixed hyperparameters, we perform 100 macro simulations for $N_{tot} = 10^5$ and take the median over the MLE estimates; the fitted values are $\theta_{S^1} = 16.08261, \theta_{S^2} = 51.51801, \sigma^2 = 309432.3$. To further stabilize the GP prediction, we fix its trend function to be the intrinsic value of the portfolio, that is,

$$\mu(z^1, z^2) = 100e^{-0.04}(z^1 - 50)_+ - 50e^{-2\cdot 0.04}(z^2 - 85)_+ \tag{5.36}$$

in Equation (5.12).

After the first stage, we narrow the candidate set to

$$\{z^m \in \mathcal{Z} : W_1(z^m)/(\sum_{n=1}^{N} W_1(z^n)) > 10^{-10}\}$$

and

$$\{z^n \in \mathcal{Z} : W_1^{\text{TVaR}}(z^n) > 10^{-10}\}$$

for TVaR. As an illustration, one run of this reduction yielded the candidate set to be narrowed from 10000 scenarios to 528 (657 for TVaR), resulting in a reduction of a factor of ≈ 20. Note that this is only after the first stage – further narrowing can be done in later stages if desired.

A visualization of the various weights discussed are given in Figure 5.1. Both panels illustrate convergence of the true probabilities $w_k^n = \mathbb{P}(z^n \in \mathfrak{R}|\mathcal{D}_k)$ as k increases from 10 to 100. Additionally, we see the shapes of ϕ_k^n and Φ_k^n, the normalized versions of $W_k(z^n; \hat{R}_k) = \phi(m_k(z) - \hat{R}_k, s_k^2(z) + \varepsilon^2)$, and $W_k^{TVaR}(z; \hat{R}_k) =$

$\Phi(m_k(z) - \hat{R}_k, s_k^2(z) + \varepsilon^2)$ respectively, become more well defined as k increases. Various spikes occur when $k = 10$, representing locations that have not been searched. The Harrell-Davis weights closely match the true GP probabilities, much moreso than ϕ_k^n and Φ_k^n. Note that the Harrell-Davis weights do not change over time; they are fixed with respect to k.

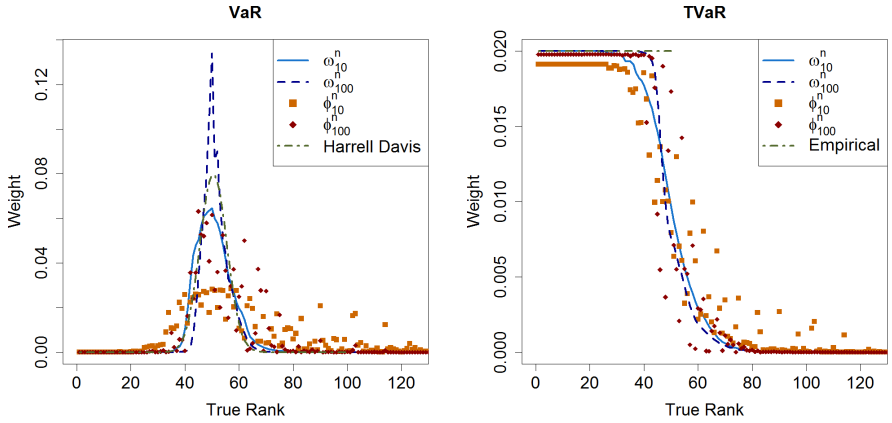

Figure 5.1: Comparing various weights defined after stages $k = 10$ and $k = 100$ for the ST-GP method. Weights $w_k^n = \mathbb{P}(z^n \in \mathfrak{R}|\mathcal{D}_k)$ the true GP probability of $z \in \mathfrak{R}$ (approximated via 10^5 simulations), and ϕ_k^n and Φ_k^n are the is the normalized versions of $W_k(z^n; \hat{R}_k) = \phi(m_k(z) - \hat{R}_k, s_k^2(z) + \varepsilon^2)$, and $W_k^{TVaR}(z; \hat{R}_k) = \Phi(m_k(z) - \hat{R}_k, s_k^2(z) + \varepsilon^2)$ respectively. Also plotted are the weights used for the risk measure, which are Harrell-Davis for VaR and $\frac{1}{50}$ for TVaR.

Comparing Algorithms

To compare the algorithms, we compute the Bias and Variance of the respective R estimators. To do so, we perform 100 macro-replications $m = 1, \ldots, 100$ with fixed outer scenario set $\mathcal{Z}$ for both $\text{VaR}_{0.005}$ and $\text{TVaR}_{0.005}$, yielding estimates $R^1, \ldots, R^{100}$. We then set $\overline{\text{bias}} \doteq mean((R_K^{1:100}) - R_{BS}$, $\overline{s} \doteq sd(R^{1:100})$ and $RMSE \doteq mean((R_K^{1:100} - R_{BS})^2) = \overline{bias}^2 + \overline{s}^2$. This yields a true sampling

distribution of the estimators from various algorithms, controlling for the intrinsic variability of inner simulations.

Figures 5.2 and 5.3 show boxplots of the resulting distributions for $m(R_K^{\text{VaR}})$ and $s(R_K^{\text{VaR}})$ ($m(R_K^{\text{TVaR}})$ and $s(R_K^{\text{TVaR}})$ for TVaR) , where the horizontal line is the true risk measure R_{BS} (R_{BS}^{TVaR} for TVaR) obtained from the analytic Black-Scholes computation. Table 5.4 reports average bias, estimator standard deviation, average squared error, and time taken over each run.

	$\text{VaR}_{0.005}$			
	$\overline{\text{bias}(R_K^{\text{VaR}})}$	$\overline{s(R_K^{\text{VaR}})}$	$RMSE$	Time (s)
ST-GP	11.85	59.64	52.04	67.8764
SE-GP	32.66	64.48	74.3	70.2592
SV-GP	36.91	50.75	64.38	86.9793
U1-GP	-1124.09	193.35	1183.93	61.95
SR-GP-1	77.59	68.97	110.39	51.8262
U2-GP	181.6	78.7	200.29	7.508
SR-GP-2	68.02	52.5	87.31	62.183
A3-GP	91.05	62.1	112.17	9.1508
True-SA	0.7	46.39	43.51	5.13
	$\text{TVaR}_{0.005}$			
	$\overline{\text{bias}(R_K^{\text{TVaR}})}$	$\overline{s(R_K^{\text{TVaR}})}$	$RMSE$	Time (s)
ST-GP	51.9	60.43	76.82	93.9
SV-GP	80.46	57.98	98.25	84.4178
U1-GP	-652.29	260.87	775.08	63.695
U2-GP	208.26	99.78	237.09	7.1759
SR-GP-2	104.48	70.2	127.7	41.2674*
A3-GP	113.12	66.09	128.34	8.498

Table 5.4: For the option portfolio case study, average values over 100 macro replications of bias, standard deviation, and squared error of R_k for each approach, as well as average time taken to complete the procedure (in seconds). Rows are missing for $\text{TVaR}_{0.005}$ when a method is not applicible. Description of methods are provided in Table 5.1. *A3-GP TVaR_α only uses 50 rounds compared to all other sequential methods which use 100 due to requiring $r_k^n > 1$ for calculating $\hat{\tau}_k^2(z^n)$ in Equation (5.15).

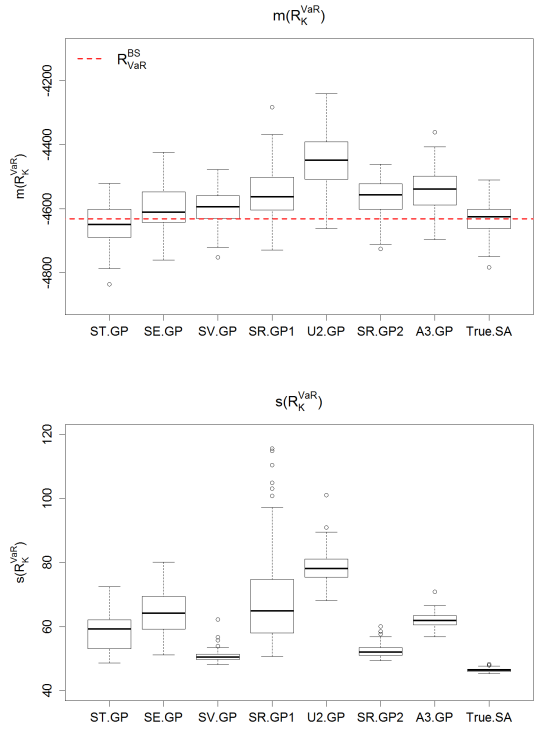

Figure 5.2: Boxplots of final $\hat{R}_K^{\text{VaR}}$ estimates and corresponding uncertainty $s(\hat{R}_K^{\text{VaR}})$ over 100 macro replications for each approach.

The uniform design (U1-GP) results are not included in the boxplots since its values are far beyond the boundaries for other methods. As mentioned, SE-GP, SR-GP-1 and True-SA are not included in $\text{TVaR}_{0.005}$ since they cannot be easily modified to the general tail instead of the exact level.

Under this budget of $N_{tot} = 10^4$, the tables illustrate complete failure of the traditional nested Monte Carlo method (U1-GP), reporting an average bias of -1124.09 (-1070.78 for TVaR), even with the kriging model for assistance. On the other hand, U2-GP, which is a simple two-stage adaptation of MC with reduction to a candidate set after a cheap initial search, provides a remarkable improvement, producing an average bias of 181.6 and RMSE of 200.29 (208.26 and 40.933 for

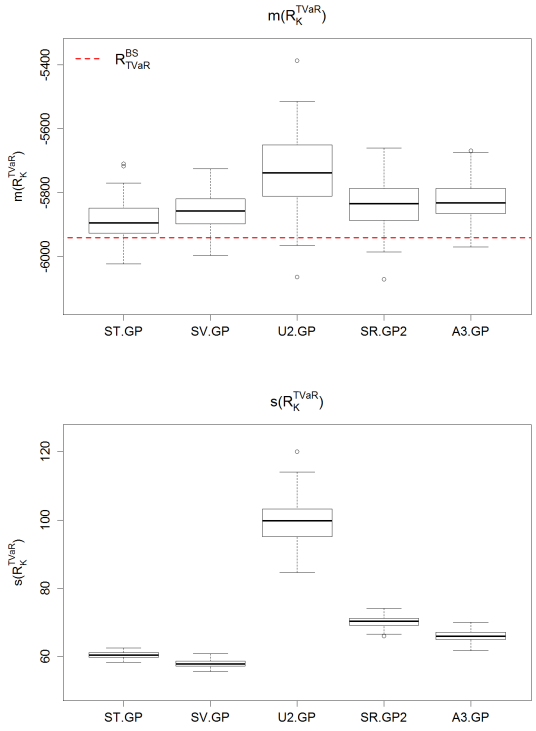

Figure 5.3: Boxplots of final $\hat{R}_K^{\text{TVaR}}$ estimates and corresponding uncertainty $s(\hat{R}_K^{\text{TVaR}})$ over 100 macro replications for each approach.

TVaR), a rough 500% improvement over all comparisons. The relatively poor performance of SR-GP-1 to other sequential methods (average bias 77.59 for VaR) indicates need for searching beyond the current guess for $\hat{z}$. Looking at both the tables and figures, the conservative sequential benchmark SR-GP-2 outperforms all non-sequential methods, suggesting that some sort of sequential algorithm is necessary in this type of problem.

Comparing the SUR methods, we see that explicitly defined uncertainty criteria gives a large improvement over other methods. Focusing on the $\text{VaR}_{0.005}$ results, SE-GP and SV-GP have approximately a 2.5x bias reduction over the next best benchmark (ignoring True-SA), and ST-GP has a 6x bias reduction. In

general, ST-GP is less effective than SE-GP, due to the fact that SE-GP does not account for uncertainty in the level for its improvement criteria. We also notice that SV-GP has the lowest average value for $s(\hat{R}_K)$; this is due to the fact that its improvement criteria is defined explicitly to reduce estimator variance. Looking at the best possible benchmark True-SA, which allocates all replications to the correct scenario, the methods provide reasonable standard deviation values, with ST-GP, SE-GP and SV-GP having values of 59.64, 64.48, 50.75 and True-SA having 46.39. Thus even under a relatively small budget of $N_{tot} = 10^4$, the SUR algorithms achieve variances nearly equal to the best possible budget allocation method.

Lastly, we provide figures illustrating evolution of the estimated risk measure $m(R_k)$ for the sequential algorithms as budget is spent. This is one of the major advantages of a sequential procedure which allows "online" use of the algorithm. Thus the user can monitor $m(R_k)$ for example to judge the convergence, adaptively stop the simulations, or report interim estimates.

Figures 5.4 and 5.5 are fan plots showing quantiles over macro replications of $m(R_k)$ as a function of k. First, the kink at $k = 10$ for A3-GP is due to fitting only considering scenarios with $r_k^n \geq 10$ for more stable values of $\hat{\tau}_k^2(z^n)$; this threshold is met for a large quantity of points when $k = 10$. The fan plots illustrate many major features of each sequential method. First, we see that SV-GP, which is defined to decrease estimator variance, has the least varying estimates in general. Sacrificed is its ability to reduce bias, where ST-GP and SE-GP achieve a lower bias more quickly. We see that SR-GP-1 takes much longer to converge to a bias of 0, since this method greatly weighs against searching in unexplored areas.

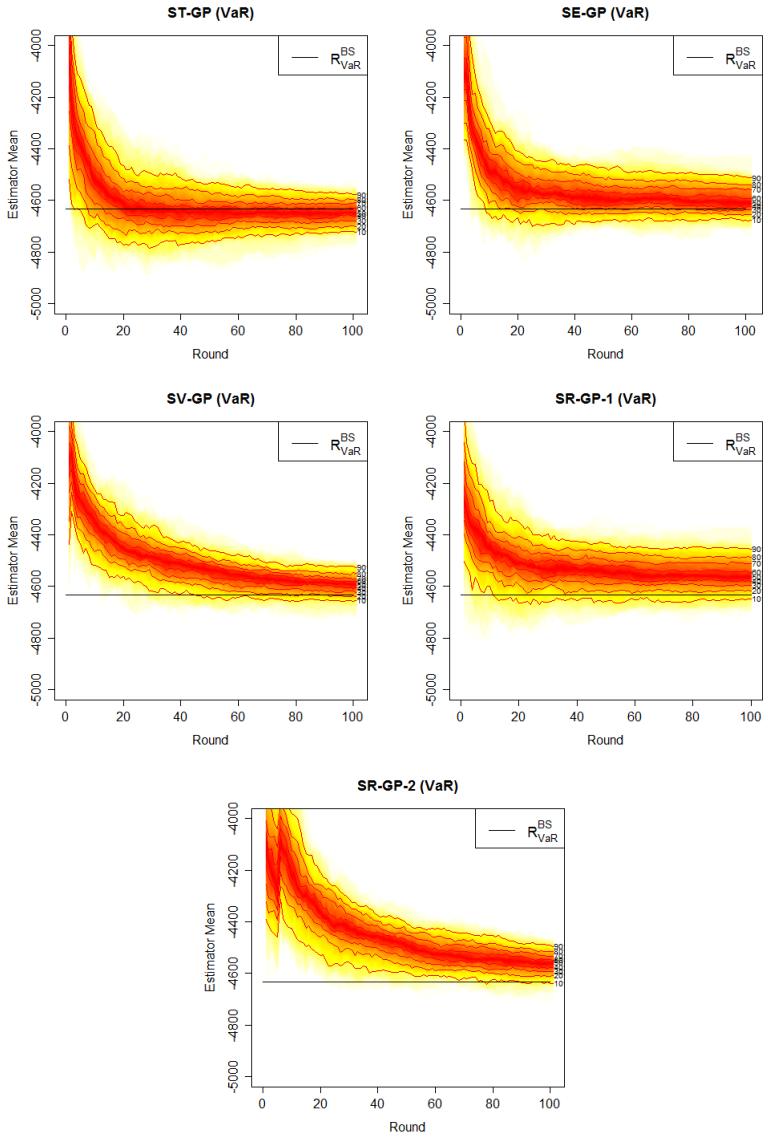

Figure 5.4: For the option portfolio case study, fan plots dscribing evolution of $m(R_k^{\text{VaR}})$ in Equation (5.19) as budget is spent. Shown are various quantiles of $m(R_k^{\text{VaR}})$ over the 100 macro repliactions, for each k.

Throughout all of the plots, we do see convergence to a bias of 0, as well as the uncertainty decreasing over time. This is an indication that the sequential methods are consistent as k increases.

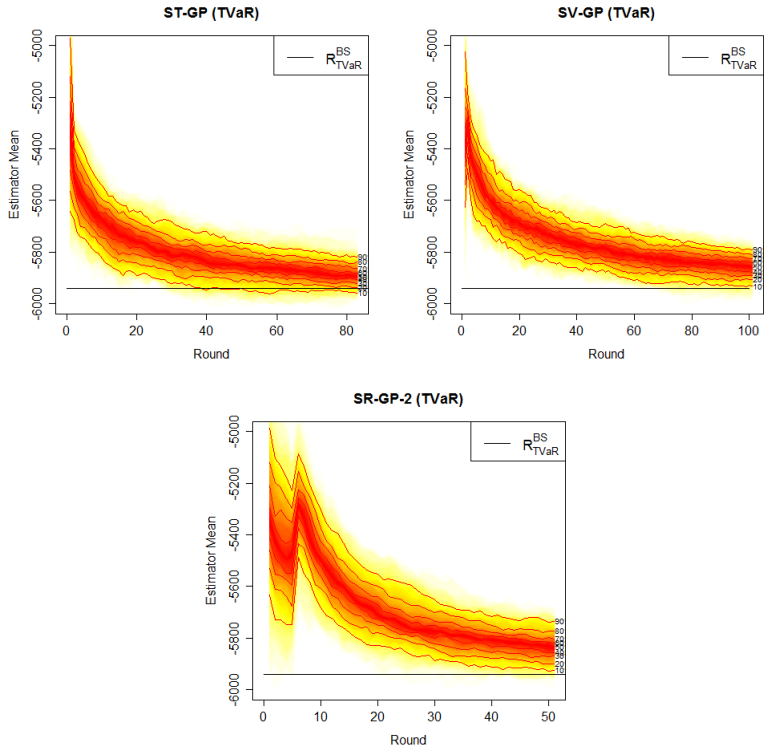

Figure 5.5: For the option portfolio case study, fan plots describing evolution of $m(R_k^{\text{TVaR}})$ in Equation (5.19) as budget is spent. Shown are various quantiles of $m(R_k^{\text{TVaR}})$ over the 100 macro repliactions, for each k.

Gains from Spatial Modeling

We next consider the improvement for learning the risk measure thanks to the spatial borrowing of information across scenarios, ignoring sequential budgeting.

We illustrate the performance of the Monte Carlo estimates for various budgets in Table 5.5. U1-GP fits a GP to the lowest 2500 sorted Monte Carlo averages over $\mathcal{D}_1$. The difference in time for U1-SA and U1-GP is roughly 58 seconds – the overhead time required to fit the 2500 scenarios and predict on $\mathcal{D}$. Above, we find that a small value of $N_{tot} = 10^4$ still quickly indicates convergence of the procedures. Note that this budget implies, for the pure Monte Carlo case,

an absurd budget of $r^n = 1$ for each $n = 1, \ldots, N$, not even allowing a variance estimate. Comparing to Table 5.4, we see that it requires an unreasonable budget to match the performance of any sequential estimator.

	U1-SA			
N_{tot}	$\overline{\text{bias}(R_k)}$	$\hat{\tau}_1(\hat{z}_1)$	$RMSE$	Time (s)
$1 \cdot 10^4$	-6830.31	NA	6844.57	3.604818
$2 \cdot 10^4$	-3768.51	1256.7	3783.02	6.956035
$5 \cdot 10^4$	-1679.35	659.39	1689.82	17.62843
$1 \cdot 10^5$	-942.88	419.91	953.17	35.34391
$2 \cdot 10^5$	-498.04	281.63	506.19	69.70281
$5 \cdot 10^5$	-250.8	168.81	262.5	169.3209
$1 \cdot 10^6$	-155.85	116.48	170.33	338.213
	U1-GP			
N_{tot}	$\overline{\text{bias}(R_k)}$	$\overline{s_1(R_1)}$	$RMSE$	Time (s)
$1 \cdot 10^4$	-1063.99	205.72	1098.84	61.567
$2 \cdot 10^4$	555.64	107.05	667.75	65.2391
$5 \cdot 10^4$	1029.24	118.2	1048.33	75.8528
$1 \cdot 10^5$	685.94	101.75	697.61	93.6108
$2 \cdot 10^5$	346.39	84.38	359.11	127.6194
$5 \cdot 10^5$	131.15	62.11	147.36	227.2054
$1 \cdot 10^6$	62.85	47.7	74.46	396.2424
	BR-SA			
N_{tot}	$\overline{\text{bias}(R_k)}$	$\overline{s_1(R_1)}$	$RMSE$	Time (s)
$5 \cdot 10^4$	-125.27	214.12	166.92	112.4932
$1 \cdot 10^5$	-70.09	217.67	88.79	145.221

Table 5.5: For the option portfolio case study, average values over 100 macro replications of bias, posterior standard deviation, and $RMSE$ of the final VaR estimator for Monte Carlo simulations, as well as average time taken to complete the procedure (in seconds). $N_{tot} = 10000$ is the actual case study budget. U1-SA is estimation through pure Monte Carlo sample averages, and U1-GP uses a GP fitted to the resulting Monte Carlo data, and BR-SA is the adaptive algorithm in Broadie et al. (2011).

On the contrary, BR-SA does much better than these methods, though still requiring a larger budget to produce desirable results. Here, $N_{tot} = 10^4$ and

$N_{tot} = 2 \cdot 10^4$ are the same as U1-SA, since the algorithm always initially assigns $r^n = 1$ to each z^n, and also it requires an estimate of $\tau^2(z^n)$, requiring $r^n \geq 2$ at each location. For any larger budget, the actual sequential algorithm kicks in.

5.6.2 Comparison of SUR Criteria

The three main methods used are ST-GP, SE-GP, and SV-GP. In general we found above that the performances of ST-GP and SE-GP are nearly identical, with SE-GP having slightly higher variance, likely due to it not taking into account uncertainty of the level. SV-GP reported the lowest variance, unsurprisingly since it is designed to minimize estimator variance, but it sacrifices bias in order to do so.

We provide a more detailed analysis comparing these three, through the use of three figures. These figures are done over one round and provide analysis on replication amounts, as well as plots comparing posterior standard deviation and mean, all versus true rank. In particular, the first comparison is through Figure 5.6, comparing the (log) replication amounts over a single run for both VaR and TVaR.

First note that the lowest values on this plot are equal to log(10), not 0; these points are ones that received only one pick from ST-GP and SE-GP, and those that were in the initialization round for SV-GP. We see that the allocation strategies for ST-GP and SE-GP are quite similar, reinforcing the conclusion that they behave similarly in general. SV-GP behaves quite differently, attaching replications of varying degree all throughout. This is due to how it handles batching: when these criteria decide how to allocate, it weights importance on various points. ST-

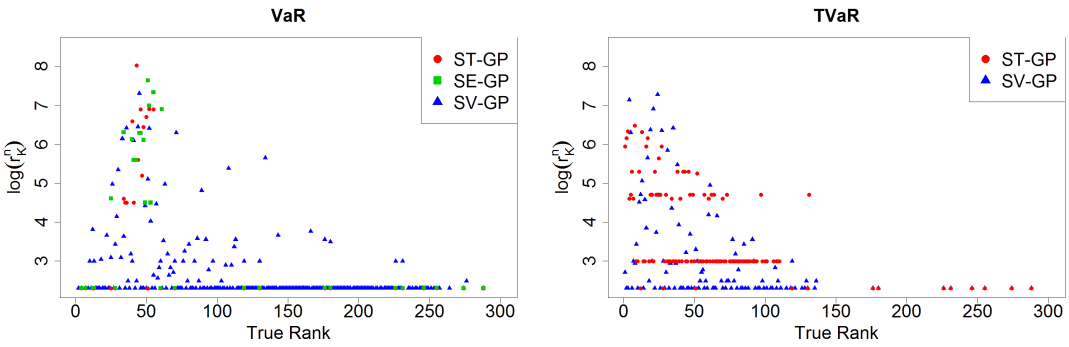

Figure 5.6: Comparing replication amounts versus true rank of $f(z)$ after the final stage for sequential methods. The y-axis is on the log scale.

GP and SE-GP attach $\Delta r = 100$ to the point they deem most important, even though many points could potentially have acquisition function values of similar magnitude, only slightly less than the most important scenario. On the other hand, SV-GP can distribute these values essentially according to importance, so in a single round there are several scenarios receiving small values of r'^n_k, and generally one or two receiving larger values. The TVaR plot tells a different story. Here, ST-GP attaches interest to points deeper in the tail, so that it needs to explore more areas rather than only around the local estimate for VaR. Interestingly, ST-GP and SV-GP behave somewhat similarly in how they taper off almost linearly from true rank 0 to 125 (roughly).

Next is Figure 5.7. This observes the final values at $k = 100$ for one run of the experiment for mean, along with the observed $\bar{y}^n_K$ and credible intervals. Around each $\bar{y}^n_K$ is a 95% confidence interval using the intrinsic noise estimate $\hat{\tau}^2_K(z^n)$. First analyzing SE-GP, we see much smaller error bars in the neighborhood of rank 50, as this is where more allocations are given. Locations with large error bars are likely spatially distant from the true VaR, where the algorithm discovered

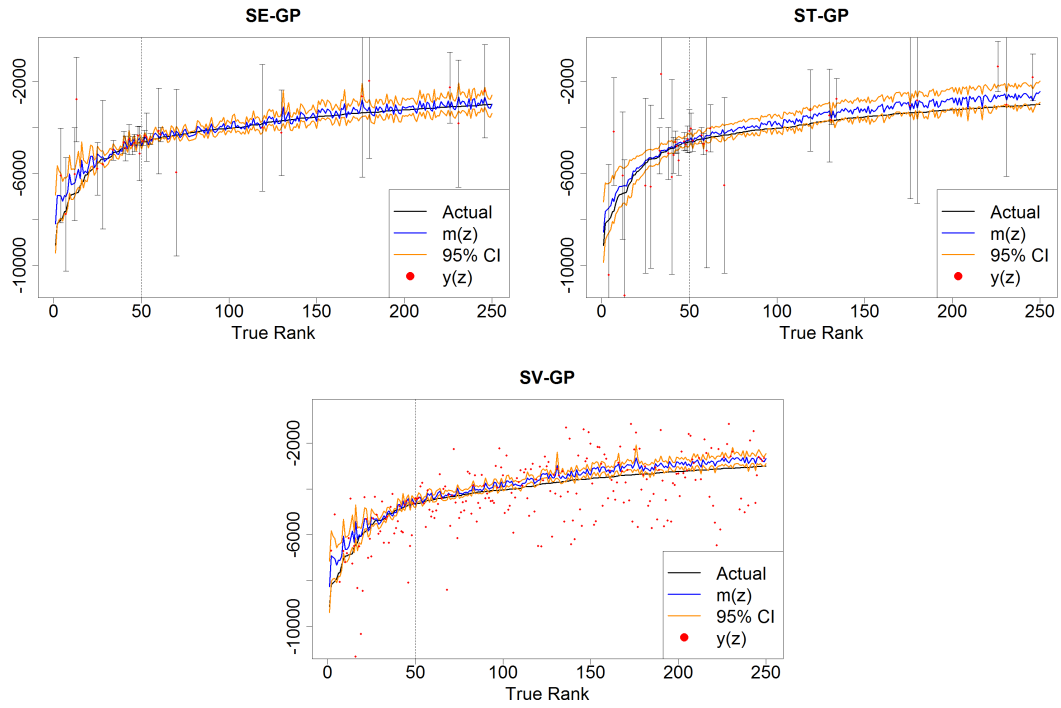

Figure 5.7: After $K = 100$, posterior mean $m(z)$, along with 95% credible intervals, the true output according to $f(z)$, as well as the observed sample averages $\bar{y}_K^n$ sorted according to the true rank of $f(z^n), n = 1, \ldots, N$, for one run of the experiment. Shown for the first two plots are 95% confidence intervals around $\bar{y}_K^n$ according to the intrinsic noise estimate $\hat{\tau}_K^2(z^n)$. These are left off to reduce clutter for SV-GP since too many scenarios are chosen.

it was not worth pursuing after a few replications were added. Next, we can see how different the 95% credible intervals using $s_K(z^n)$ are versus $\hat{\tau}_K^2(z^n)$; only in regions where many replications were added are where $s_K(z^n)$ and $\hat{\tau}_K^2(z^n)$ are both small. Regardless this plot shows how $s_K(z^n)$ smoothes out observation error. We also see in all three plots that it does a poor job near the extreme left tail and toward the right tail, especially with ST-GP and SV-GP, and that all three approach the true $f(z)$ as the true rank nears 50, illustrating its accuracy due to more replications near rank 50 and less toward the tails. Additionally, the

credible intervals are wider at these tails.

Comparing the plots, we observe that SE-GP and ST-GP again produce similar results, with the differences possibly due to errors over one run of the experiment. The smaller variances of SV-GP are apparent, yielding smaller credible intervals nearly uniformly. In addition, we see the large number of points picked for this method, with the observations $\bar{y}_K^n$ scattered all throughout. Contrasting, SE-GP and ST-GP have much fewer values of $\bar{y}_K^n$ reported; this is because $\bar{y}_K^n$ has no reported value if no replications are added to z^n.

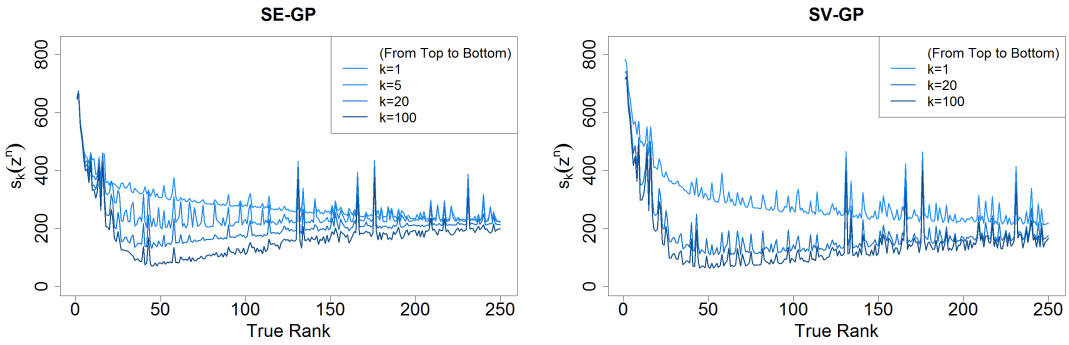

Figure 5.8: Posterior standard deviation $s_k(z)$ reported over multiple rounds for one run, sorted according to the true rank of $f(z^n), n = 1, \ldots, N$.

Lastly, Figure 5.8 illustrates evolution of $s_k(z)$ for $k = 1, 5, 20, 100$. Left out is ST-GP, which matches SE-GP nearly identically over each round. Both plots show similar behaviors, with $s_k(z)$ decreasing uniformly in z as k increases. We see that it decreases much more near true rank 50, and that it decreases very little at both tails, a desirable feature to have in this type of algorithm.

5.6.3 Comparing GP Methods

We compare choice of prior kernel (Matern 5/2 versus Gaussian) and noise modeling technique. The package "DiceKriging" uses point estimates $\hat{\tau}^2(z^n)$ to estimate $\tau^2(z^n)$, while "hetGP" uses a separate model for the noise surface $\tau^2(\cdot)$. The package also computes likelihood for the model where the noise surface is homogeneous, i.e. $\tau^2(z) \equiv \tau^2$ in case this is found to be more optimal. As in the previous sections, table 5.6 provides average bias, posterior standard deviation, and RMSE over 100 macro replications, and Figure 5.9 illustrates evolution as budget is spent through fan plots over 100 macro replications.

`R` package	Kernel	bias($\hat{R}_K$)	$s(\hat{R}_K)$	$RMSE$	Time (s)
DiceKriging	Matern 5/2	-16.81	58.38	61.43	109.4093
hetGP	Matern 5/2	-37.67	61.48	75.29	174.1756
hetGP	Gaussian	-15.45	68.77	66.68	172.9394

Table 5.6: Bias, uncertainty, and RMSE estimates over 100 macro replications comparing `R` packages and covariance kernels for the ST-GP method. The main difference is that "hetGP" fits a GP surface to estimate the noise process $\tau^2(z)$, while "DiceKriging" uses the point estimates $\hat{\tau}^2(z^n)$ as in Equation (5.15).

The table indicates no real difference between the three methods. Introducing noise modeling and still using Matern 5/2 introduces slight bias, though this could simply be error over the replications. The figure shows more interesting results: it appears that introducing a noise surface helps to eliminate positive bias after the first few rounds, however, in correcting this, it introduces a more significant negative bias that itself needs to be corrected. The variance over all methods are relatively similar.

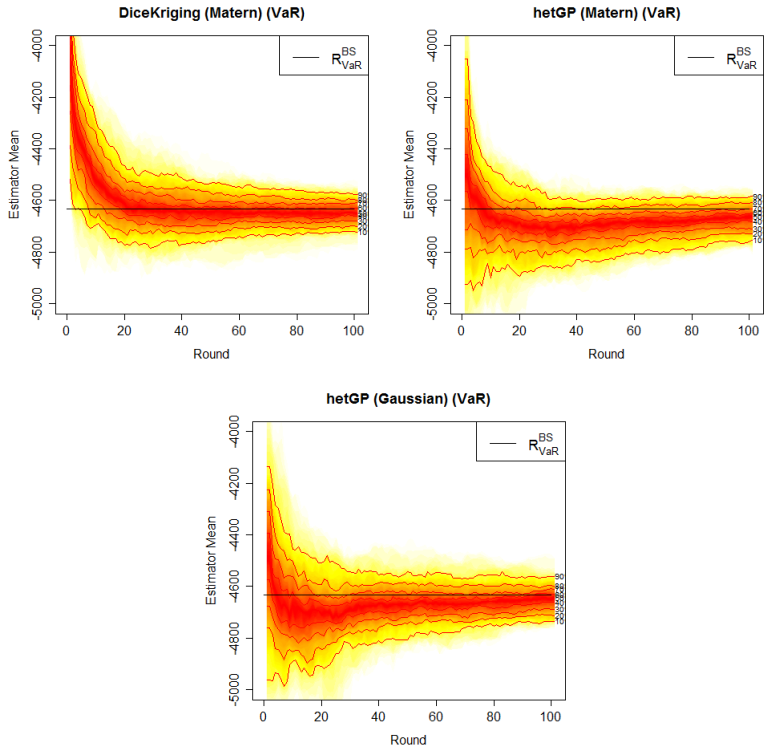

Figure 5.9: Fan plots comparing `R` packages and covariance kernels for the ST-GP method. The main difference is that "hetGP" fits a GP surface to estimate the noise process $\tau^2(z)$, while "DiceKriging" uses the point estimates $\hat{\tau}^2(z^n)$ in Equation (5.15).

5.7 Case Study: Life Annuities under Stochastic Interest Rate and Mortality

In this case study we move to a more complex example with larger dimension. The main goal is to see if the major results of Section 5.6 remain true in a larger dimension setting. We consider a setup where an annuitant receives a contract to begin payments in T years, whence the payments continue until death of the individual. In practice some cutoff age x_u is set for the final payment if the individual lives that long. As before, regulations require analysis of quantiles

of the one-year potential loss on this contract. Several factors affect the value and length of such a contract, especially interest rate and mortality risk. We emulate this by introducing two three factor models, one for interest rate and one for mortality probability evolution. The result is a six-factor GP, and the continuous time nature of the models along with longer maturity also yields a more computationally expensive simulator, reducing the impact of overhead cost in fitting, selection, and prediction.

To provide the mathematical details, let β_t be instantaneous interest rate at time t, and $\tau(x)$ be the remaining lifetime of the individual. Assume a Markov state process (explicitly defined later) (Z_t) for $(\mathcal{F}_t)$. We introduce the notations, for $u \leq t < T$

$$\mathbb{P}(\tau_x > T | \tau_x > t, \mathcal{F}_u) \doteq P(Z(u); t, T, x) \tag{5.37}$$

$$q(Z(u); t, T, x) \doteq P(Z(u); t, T, x) - P(Z(u); t, T+1, x) \tag{5.38}$$

to be the probability that an individual aged x at time t survives to time T, given the information at time u, and the probability that an individual aged x at time t dies between years T and $T+1$, given the information at time u, respectively.

If T is the date at which payments begin, then conditioning on $\tau > t$ and the information $\mathcal{F}(t)$ available by time t, and assuming that accrued payments by t are normalized to be 1, the present value of this annuity at time t, assuming

$t \leq T$, is

$$f(z) = \sum_{j=T}^{\infty} \mathbb{E}\left[e^{-\int_t^j \beta_u du} 1_{\{\tau(x+t)\geq j\}} \middle| Z_t = z \right]$$
$$= \sum_{j=T}^{\infty} \mathbb{E}\left[e^{-\int_t^j \beta_u du} \middle| Z_t = z \right] P(Z(t); t, j, x+t). \tag{5.39}$$

To precise the modeling of (Z_t) we blend the setups of Chen (1996) and Cairns et al. (2011a). The interest rate dynamics for (β_t) is defined through a three-factor Cox-Ingersoll-Ross model with stochastic volatility ζ_t and stochastic mean-reversion level α_t

$$d\beta_t = (\bar{\beta} - \alpha_t)dt + \sqrt{\beta_t}\zeta_t dW_t^{\beta}, \tag{5.40}$$
$$d\alpha_t = (\bar{\alpha} - \alpha_t)dt + \sqrt{\alpha_t}\zeta_t dW_t^{\alpha},$$
$$d\zeta_t = (\bar{\zeta} - \zeta_t)dt + \sqrt{\zeta_t}\varphi dW_t^{\zeta}$$

(5.41)

where W^{β}, W^{α} and W^{ζ} are independent standard Brownian motions.

For details on mortality modeling, we defer the reader to Cairns et al. (2011a). The `R` package StMoMo Villegas et al. (2015a) contains England & Wales (E&W) mortality data as well as tools for fitting and simulating the models in Cairns et al. (2011a). For easy accessibility, we use the E&W data and choose model (M7) from Cairns et al. (2011a), defined as follows,

$$\operatorname{logit} q(Z_u; t, t+1, x) = \kappa_t^1 + \kappa_t^2(x - \bar{x}) + \kappa_t^3\left((x - \bar{x}^2 - \hat{\sigma}_x^2\right) + \gamma_{t-x}, \tag{5.42}$$

where $\bar{x}$ is the average age the model is fit to, and $\hat{\sigma}_x^2$ is the mean value of $(x-\bar{x})^2$ are interpreted as *age* effects, the processes $\kappa_t^i \equiv \kappa^i(t, Z_u)$ are the *period* effects capturing mortality evolution over calendar year, and $\gamma_{t-x} \equiv \gamma(t-x, Z_u)$ is the *cohort* effect. By simulating this process, we can back out the $P(Z(t); t, j, x+t)$ needed for Equation (5.39) via Equation (5.38).

This implies a Markov state process of $Z_t = (\beta_t, \alpha_t, \zeta_t, \kappa_t^1, \kappa_t^2, \kappa_t^3)$. The one-year future value of an annuity then is $f(Z_1)$ in Equation (5.39). As before, we simulate Z_1 via Algorithm 2 to determine a scenario set $\mathcal{Z}$ with $N = 10^4$, and investigate $\text{VaR}_{0.005}$ and $\text{TVaR}_{0.005}$, the $0.005-$quantile and tail average of $\{f(z) : z \in \mathcal{Z}\}$.

The interest rate model is a continuous time model, and we perform a simple forward Euler method with discretization $\Delta t = 0.1$. Here, $f(\cdot)$ takes approximately 0.01115 seconds to evaluate, while it takes 0.000513 seconds to evaluate for the first case study. In industry, the evaluator typically takes $f(\cdot)$ significantly longer. This all implies that the numeric overhead of fitting and predicting becomes negligible as the model becomes more realistic.

5.7.1 Results

For the remainder of this section we analyze $f(Z_1)$, the net present value of the life annuity one year into the future. We let $x = 55$ and the expiration time be $T = 10$. The interest rate parameters in Equation (5.40) are parameters $\bar{\beta} = 0.04, \bar{\alpha} = 0.04, \bar{\zeta} = 0.02, \phi = 0.05$, with the mortality model fitted over the age range $x \in [55, 89]$ using the StMoMo package in `R` Villegas et al. (2015a). The GP has hyperparameters $\theta_\beta = 8.9668, \theta_\alpha = 13.0322, \theta_\zeta = 12.5858, \theta_{\kappa^1} = 13.7758, \theta_{\kappa^2} = 13.0073, \theta_{\kappa^3} = 15.1429$, and $\sigma^2 = 2.127$; these θ values are based

on standardized inputs by subtracting mean and dividing by standard deviation, marginally in each dimension.

Our aim is to repeat the analysis in Section 5.6 to see how the results extend to a more realistic higher dimension example. Under the setup (5.39), there is no closed form evaluation for $f(z)$, so we obtain a benchmark through simulation – this value is determined by performing ST-GP and SR-GP-2 in alternating rounds with a budget of $2 \cdot 10^7$, a budget 2000x larger than that of a single macro replication in our experiment. The result is $R_B^{\text{VaR}} = -16.0529$, $R_B^{\text{TVaR}} = -16.37795$ with estimator standard deviations of $s(R_B^{\text{VaR}}) = 0.002017$ and $s(R_B^{\text{TVaR}}) = 0.001879$.

We repeat the methods in Section 5.6, performing 100 macro replications with fixed $\mathcal{Z}$ for both $\text{VaR}_{0.005}$ and $\text{TVaR}_{0.005}$. The box plots for bias and squared error are reported in Figures 5.10 and 5.11, with fan plots in 5.12 and 5.13, and the numeric values for average bias, estimator standard deviation, squared error, and time taken are reported in Table 5.7.

Remarkably, the results mirror those of the first case study almost perfectly in terms of relative order of sequential methods and benchmark performance, with relative improvements for the SUR results in most cases. To provide details, ST-GP retains the lowest bias, but here its bias is roughly 29x lower than that of U2-GP, the simple two stage procedure, compared to 15x lower in the first case study, and now its average time spent for one run is only 2.5x as long compared to 9x in the first case study, echoing that the numeric overhead decreases as $f(\cdot)$ becomes more expensive to evaluate. The relative performance of ST-GP versus A3-GP, the three-stage benchmark based on variance minimization, is similar to that of case study 1. We also find that A3-GP does better than in case study 1

	$\text{VaR}_{0.005}$			
	$\overline{\text{bias}(R_K^{\text{VaR}})}$	$\overline{s(R_K^{\text{VaR}})}$	$RMSE$	Time (s)
ST-GP	0.0227	0.0549	0.0689	287.9145
SE-GP	0.027	0.0542	0.0565	312.8884
SV-GP	0.0651	0.0419	0.0756	414.0081
U1-GP	-2.701	0.5505	2.704	177.1371
SR-GP-1	0.0764	0.0455	0.093	188.3068
U2-GP	0.6612	0.2151	0.691	111.5368
SR-GP-2	0.0715	0.0416	0.0821	226.0427
A3-GP	0.1115	0.0449	0.1203	114.2593
	$\text{TVaR}_{0.005}$			
	$\overline{\text{bias}(R_K^{\text{TVaR}})}$	$\overline{s(R_K^{\text{TVaR}})}$	$RMSE$	Time (s)
ST-GP	0.0036	0.0583	0.0658	330.1495
SV-GP	0.0628	0.0391	0.0731	403.8966
U1-GP	-2.4446	0.3559	2.4509	176.6486
U2-GP	0.8335	0.2367	0.8663	112.3129
SR-GP-2	0.0803	0.0427	0.0894	178.2559*
A3-GP	0.1098	0.046	0.1193	114.9563

Table 5.7: For the life annuity case study, average values over 100 macro replications of bias, standard deviation, and squared error of R_k for each approach, as well as average time taken to complete the procedure (in seconds). Rows are missing for $\text{TVaR}_{0.005}$ when a method not being applicible. Description of methods are provided in Table 5.1. *A3-GP TVaR_α only uses 50 rounds compared to all other sequential methods which use 100 due to requiring $r_k^n > 1$ for calculating $\hat{\tau}_k^2(z^n)$ in Equation (5.15).

compared to the other benchmarks.

Comparing the SUR algorithms, the results are similar to the first case study, though SE-GP does marginally better than ST-GP for VaR; this is argued by it having the lowest MSE as well as looking at the box and fan plots, where in general it is less varying with the same or better bias results. In general, the shape of the fan plots are also similar to case study 1, with convergence rates looking similar as well.

These results imply that sequential methods based on GP's still perform well

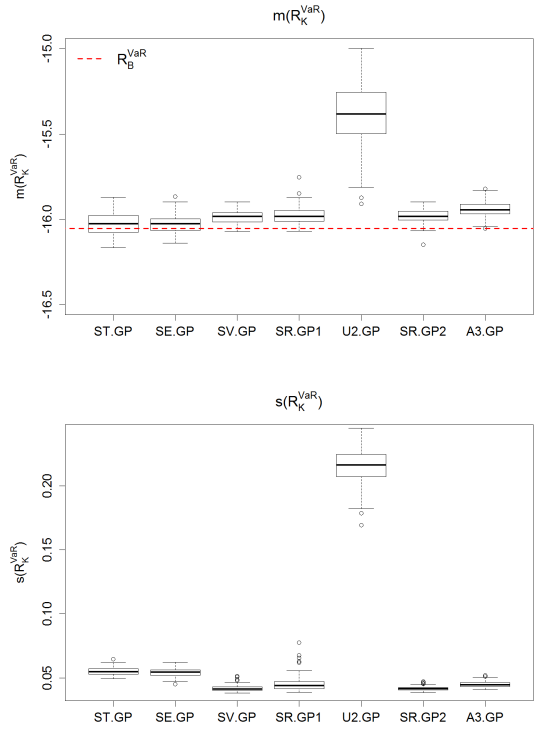

Figure 5.10: For the life annuity case study, boxplots of final $m(R_K^{\mathrm{VaR}})$ estimates and $s(R_K^{\mathrm{VaR}})$ over 100 macro replications for each approach.

in a higher dimensional setting. In fact, they perform relatively better than in the lower dimensional example.

5.8 Conclusion

We investigated performance of stepwise uncertainty reduction (SUR) algorithms for level set and contour estimation for an unknown noisy function f. In order to answer various questions about performance, they are compared against various benchmarks through two case studies. One is a simplistic two-dimensional example, and the second is more complex with six-dimensions. Though different

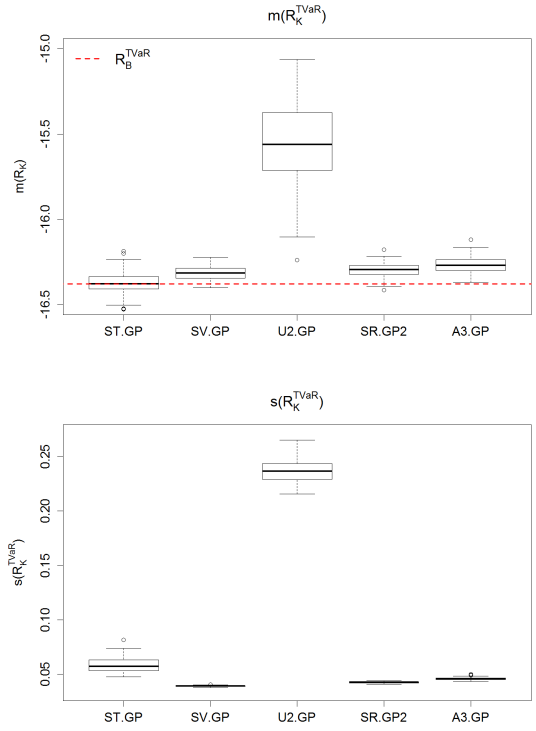

Figure 5.11: For the life annuity case study, boxplots of final $m(R_K^{\text{TVaR}})$ estimates and $s(R_K^{\text{TVaR}})$ over 100 macro replications for each approach.

in nature, both case studies yielded the same conclusions, nearly mirroring relative performance across all methods. In general, we found that even crude multi-step methods offer performance increases compared to methods involving only a few stages. In addition, the SUR methods based on explicit criteria (e.g. variance minimization of the estimator) performed much better than crude SUR benchmarks. The downside is that the numerical overhead for fitting and predicting increases with the number of stages; however, this overhead decreases from the first to second case study where calls to f become more expensive, and in practical cases, the calls to f take much longer than either case study in this paper, so that the overhead becomes negligible.

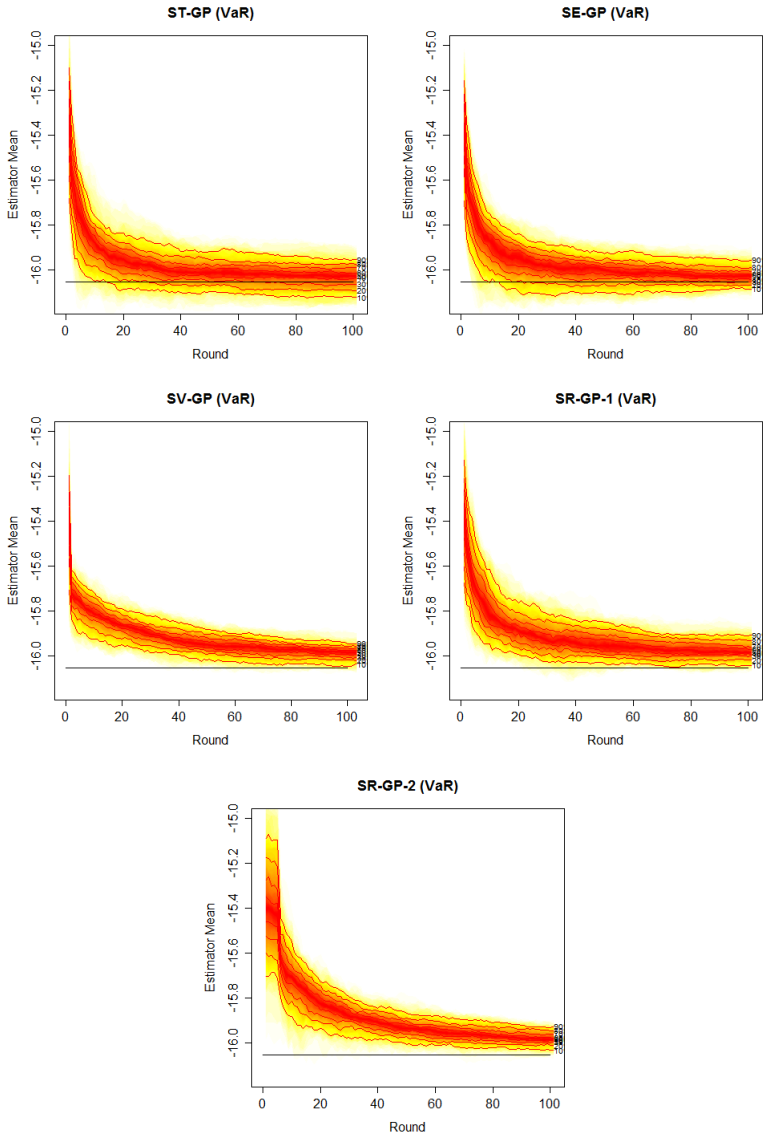

Figure 5.12: For the life annuity case study, a fan plot dscribing evolution of $m(R_k^{\mathrm{VaR}})$ in Equation (5.19) as budget is spent. Shown are various quantiles of $m(R_k^{\mathrm{VaR}})$ over the 100 macro repliactions, for each k.

The case studies analyzed VaR and TVaR, the quantile and tail average of financial loss respectively. Typically, industry uses crude Monte Carlo to estimate these values, which was by far the worst performing benchmark, even when a GP

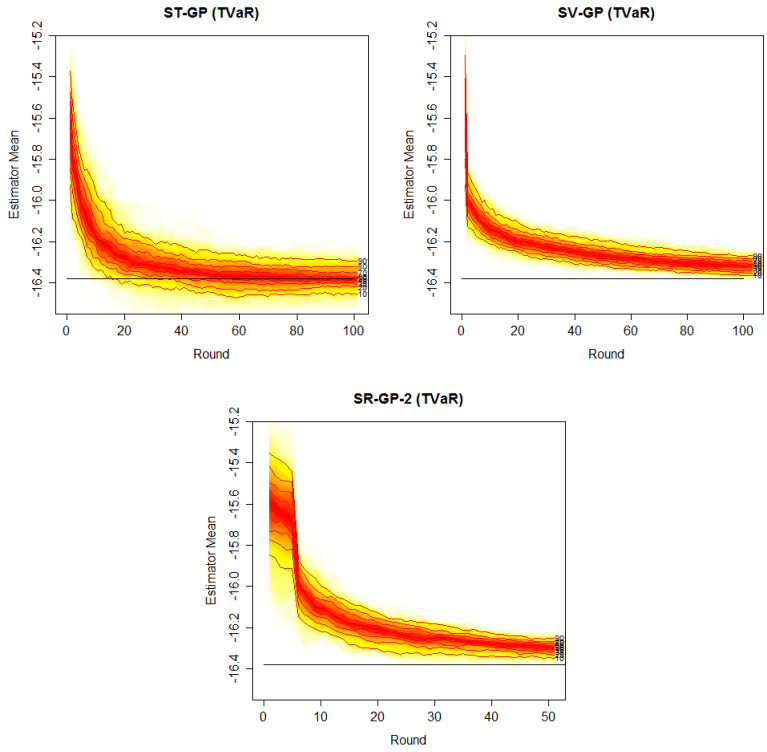

Figure 5.13: For the life annuity case study, a fan plot describing evolution of $m(R_k^{\text{TVaR}})$ in Equation (5.19) as budget is spent. Shown are various quantiles of $m(R_k^{\text{TVaR}})$ over the 100 macro repliactions, for each k.

helped interpolate the output along with optimal budgeting as suggested in Gordy and Juneja (2010). Even a simple sequential approach that breaks budgeting into two stages, where the second focuses on tail scenarios via the GP, offers a 5x bias reduction (4x in the annuity case study). The best performing algorithm yielded a 95x bias reduction (119x in the annuity case study).

This work is the first detailed analysis of level and contour estimation in the case of a noisy function, so there are many extensions to be made. First, the initial stage in each sequential algorithm was a fixed 10% of budget spent through representative points of the domain determined via a space-filling algorithm. One

can further analyze both the amount of budget spent in this step, along with the number of representative points chosen. For example, 10% may be overkill in some cases, but in others it might not provide enough insight, especially depending on the size of the domain and the number of representative points chosen. It is possible that these values should depend somehow on the problem itself, e.g. through estimator standard deviation. Additionally, the space-filling algorithm can be improved. In fact, we only care one extreme region of the domain, so an algorithm that focuses on filling extreme regions rather than the entire set of scenarios is desirable. One way to achieve this is described in Chauvigny et al. (2011), through use of statistical *depth functions*. The downside is that it requires certain assumptions on the distribution of Z_T and properties of f, though these are satisfied in most realistic scenarios.

Another extension is to consider other improvement criteria and modifications of the current SUR algorithms. For example, SV-GP minimizes the variance of the estimator, and unsurprisingly it offered the lowest posterior estimator variance. Its bias, however, was larger than the other SUR methods. Various criteria should be investigated along with theoretical optimal solutions. Furthermore, a crucial difference is that SV-GP allocates budget in a single stage to multiple points, whereas the others add multiple calls to f for a single point. One improvement is to modify the optimization for ST-GP and SE-GP to the case of a dynamic budgeting like SV-GP.

In this paper we used the empirical noise variance estimates as in Equation (5.15). This caused some difficulties in the first case study where the noise surface is extremely heteroskedastic: the scenarios producing the lowest noise were actu-

ally of unimportance, and in some runs the initialization stage resulted in a GP that gave unnecessarily large weights to these areas, resulting in poor predictions. A recent improvement to GPs in general is given in Binois et al. (2016), which give fast updating formula for a separate noise surface. GP modeling including this technique is included in an upcoming `R` package "hetGP"; the package has already been used in a disease forecasting application Johnson et al. (2017).

5.9 Appendix: Set Based Expected Improvement Criteria

See Chevalier et al. (2014b) for a more extensive overview of these criteria. We follow the nomenclature given in their paper. The goal is to understand one of

- The level set $\Gamma^* \doteq \{z \in \mathcal{Z} : f(z) \geq L\}$, where L is a fixed threshold,
- The volume of excursion $\alpha^* \doteq \mathbb{P}(\Gamma^*)$,
- The contour line $\mathcal{C}^* \doteq \{z \in \mathcal{Z} : f(z) \in (L-\varepsilon, L+\varepsilon)\}$ for small ϵ,

by sequentially picking a point to reduce uncertainty. It is worth remarking that each criteria should work well for all problems since they are so similar in nature.

5.9.1 Level Set

Denoted the *sur* criteria, we define

$$\mathcal{H}^{sur} \doteq \frac{1}{N}\sum_{n=1}^{N} p_k(z^n)\left(1 - p_k(z^n)\right), \tag{5.43}$$

where $p_k(z) \doteq \mathbb{P}(\hat{f}_k(z) \geq T | \mathcal{D}_k)$ is the probability of $\hat{f}_k(z)$ being in the level set. Intuitively, the uncertainty is low when $p_n(z)$ is close to 0 or 1 over all $\mathcal{Z}$, i.e. $\hat{f}_k$ has strong understanding of whether all points are in or not in Γ^*. The sampling criteria then aims to choose the next scenario z that best reduces this uncertainty:

$$\text{sur}(z) \doteq \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N} p_{k+1}(z^n)\left(1 - p_{k+1}(z^n)\right)\middle| z_{k+1}^{new} = z\right], \tag{5.44}$$

where the conditioning means replications are added to scenario z in the next step. Choosing the optimal scenario means evaluating Equation (5.44) at each $z \in \mathcal{Z}$ and choosing the one that results in the lowest $\text{sur}(z)$. In practice, the search domain is reduced to a smaller subset of points after the first stage, since then $\hat{f}$ can safely exclude many points far away $\tilde{z}$.

5.9.2 Volume of Excursion

The *jn* criteria is optimal for estimating $\alpha^* \doteq \mathbb{P}(\Gamma^*)$. Simply, we let $\Gamma_k \doteq \{z \in \mathcal{Z} : \hat{f}_k(z) > T\}$ and

$$\gamma_k \doteq \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{\{\hat{f}_k(z^n) > T\}} \tag{5.45}$$

is the volume of Γ_k. Then

$$\mathcal{H}^{jn} \doteq \text{var}(\gamma_k | \mathcal{D}_k), \tag{5.46}$$

and the associated sampling criterion is

$$jn(z) \doteq \mathbb{E}[\text{var}(\gamma_{k+1} | \mathcal{D}_{k+1}) | z^{new} = z). \tag{5.47}$$

5.10 Appendix: Variance Minimization Calculations

Lemma 4. *Let* $\mathbf{C} = [C(z^n, z^m)]_{1 \le n,m \le N}$ *be the covariance matrix of* $z \in \mathcal{Z}$, $\mathcal{T}_k = \left(\hat{\tau}_k^2(z^1), \ldots, \hat{\tau}_k^2(z^N)\right)$, $\boldsymbol{I}$ *be the* $N \times N$ *identity matrix, and noting that* $r_k^n = r_{k-1}^n + r_{k-1}'^n$, $\underline{\mathbf{r}}_k = \left(\frac{1}{r_{k-1}^1 + r_{k-1}'^1}, \ldots, \frac{1}{r_{k-1}^N + r_{k-1}'^N}\right)$, *Then the following is an approximation that*

improves as each r_{k-1}^n *increases:*

$$\begin{aligned}&(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_k^T)^{-1}\\ \approx&(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}+(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}(\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T-\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_k^T)(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}.\end{aligned} \tag{5.48}$$

Proof. Let $\mathbf{B}$ be the diagonal matrix with elements $\sqrt{\hat{\tau}_k^2(z^n)\left(\frac{1}{r_{k-1}^n}-\frac{1}{r_{k-1}^n+r_{k-1}'^n}\right)}$. Then by adding and subtracting,

$$\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_k^T=\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T-\mathbf{B}\mathbf{B}, \tag{5.49}$$

so that by the Woodbury matrix formula Golub and Van Loan (2012),

$$\begin{aligned}&(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_k^T)^{-1}\\ =&(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T+\mathbf{B}(-\mathbf{I})\mathbf{B})^{-1}\\ =&(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}\\ &-(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}\mathbf{B}\left(\mathbf{B}(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}\mathbf{B}-\mathbf{I}\right)^{-1}\mathbf{B}(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}.\end{aligned} \tag{5.50}$$

When r_k^n is large, both $\underline{\mathbf{r}}_{k-1}$ and $\mathbf{B}$ have relatively small entries, so that $\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T\approx\mathbf{C}$ and hence $\mathbf{B}(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}\mathbf{B}-\mathbf{I}\approx-\mathbf{I}$. Therefore,

$$(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_k^T)^{-1}=(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}+(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}\mathbf{B}\mathbf{B}(\mathbf{C}+\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T)^{-1}. \tag{5.51}$$

Plugging in $\mathbf{B}\mathbf{B}=\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_{k-1}^T-\mathcal{T}_k\boldsymbol{I}\underline{\mathbf{r}}_k^T$, finishes the proof. □

Lemma 4 provides a solution to the matrix inversion problem in Equation

(5.20).

Lemma 5. *Subject to the constraints* $\sum_{n=1}^{N} r'^{n}_{k-1} = \Delta r_{k-1}$ *and* $r'^{n}_{k-1} \geq 0$ *for* $n = 1, \ldots, N$*, the minimization problem of*

$$s^2(R_k) = \mathbf{w}_k(\mathbf{C} - \mathbf{C}(\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_k^T)^{-1}\mathbf{C})\mathbf{w}_k^T \tag{5.52}$$

with respect to $r'^{1}_{k-1}, \ldots, r'^{N}_{k-1}$ *using the approximation in Lemma 4 reduces to minimizing*

$$\mathbf{u}_k^T \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_k^T \mathbf{u}_k \tag{5.53}$$

under the same constraints, where $\mathbf{u}_k = (\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T)^{-1}\mathbf{C}\mathbf{w}_k$.

Proof. Using Lemma 4, Equation (5.52) becomes

$$\begin{aligned}
&\mathbf{w}_k(\mathbf{C} - \mathbf{C}(\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_k^T)^{-1}\mathbf{C})\mathbf{w}_k^T \\
&= -\mathbf{w}_k^T(\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T)^{-1}(\mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T - \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_k^T)(\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T)^{-1}\mathbf{w}_k \\
&= -\mathbf{w}_k^T(\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T)^{-1}\mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T(\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T)^{-1}\mathbf{w}_k \\
&\quad + \mathbf{w}_k^T(\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T)^{-1}\mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_k^T(\mathbf{C} + \mathcal{T}_k \boldsymbol{I} \underline{\mathbf{r}}_{k-1}^T)^{-1}\mathbf{w}_k.
\end{aligned} \tag{5.54}$$

The r'^{n}_{k-1} only appear in $\underline{\mathbf{r}}_k^T$, so it is equivalent to minimize this equation with respect to only the second term. □

Finally, solving the reduced problem in Lemma 5 has a closed form solution. The optimal solution can be determined using a *pegging algorithm* provided in Bretthauer et al. (1999), and is given in Algorithm 3. Note that when $\tau^2(z^n)$

is estimated, the argument r'^{n}_{k-1} appears in $\hat{\tau}^2_k(z^n)$. In general we should have $\hat{\tau}^2_{k-1}(z^n) \approx \hat{\tau}^2_k(z^n)$, so we simply use $\hat{\tau}^2_{k-1}(z^n)$ in the minimization instead. Additionally, the w^n_k are unknown, so we use w^n_{k-1}.

Algorithm 3: Algorithm to determine optimal solution to the variance minimization problem in Lemma 5.

input : $\mathcal{T}_k, \mathbf{u}_k, (r^1_{k-1}, \ldots, r^N_{k-1})$
output: $(r'^1_{k-1}, \ldots, r'^N_{k-1})$

1 Set $d^n_k \leftarrow (u^n_{k+1})^2 \hat{\tau}^2_k(z^n), n = 1, \ldots, N, \quad L \leftarrow \emptyset, \quad I \leftarrow \{1, \ldots, N\}, \quad R \leftarrow \sum_{n=1}^N r^n_k, \quad S \leftarrow \sum_{n \in I} \sqrt{d^k_n}, \quad \lambda \leftarrow (S/R)^2.$
2 Set $r'^n_{k-1} \leftarrow (d^n_k/\lambda)^2$
3 Set $x \leftarrow 0, \quad L' \leftarrow \emptyset$
4 **for** $n \in I$ **do**
5 **if** $r'^n_{k-1} < 0$ **then**
6 $x \leftarrow x - r'^n_{k-1}, \quad L' \leftarrow L' \cup \{n\}$
7 **end**
8 **end**
9 **if** $L' \neq \emptyset$ **then**
10 $I \leftarrow I \backslash L', \quad L \leftarrow L \cup L', \quad S \leftarrow S - \sum_{n \in L'} \sqrt{d^k_n}, \quad R \leftarrow R - \sum_{n \in L'} r^n_{k-1}, \quad \lambda \leftarrow (S/R)^2$. Go to Step 2.
11 **end**
12 $r'^n_{k-1} \leftarrow 0$ for all $n \in L, \quad r'^n_{k-1} \leftarrow (d^n_k/\lambda)^2$ for all $n \in I$.

Chapter 6

Gaussian Process Models for Mortality Rates and Improvement Factors

6.1 Abstract

We develop a Gaussian process ("GP") framework for modeling mortality rates and mortality improvement factors. GP regression is a nonparametric, data-driven approach for determining the spatial dependence in mortality rates and jointly smoothing raw rates across dimensions, such as calendar year and age. The GP model quantifies uncertainty associated with smoothed historical experience and generates full stochastic trajectories for out-of-sample forecasts. Our framework is well suited for updating projections when newly available data arrives, and for dealing with "edge" issues where credibility is lower. We present a detailed analysis of Gaussian process model performance for US mortality experience based on the CDC datasets. We investigate the interaction between mean and residual modeling, Bayesian and non-Bayesian GP methodologies, accuracy of in-sample and

out-of-sample forecasting, and stability of model parameters. We also document the general decline, along with strong age-dependency, in mortality improvement factors over the past few years, contrasting our findings with the Society of Actuaries ("SOA") MP-2014 and -2015 models that do not fully reflect these recent trends.

6.2 Introduction

Publishing of pension mortality tables and mortality improvement factors for use by actuarial professionals and researchers in longevity risk management is a major endeavor of the actuarial professional organizations. In the US, the Society of Actuaries (SOA) runs the Retirement Plans Executive Committee (RPEC); its most recent publication is known as the RP-2014 mortality tables and the MP-2015 improvement scales (SOA, 2014b,a). In the UK, annual tables are released in the form of the Continuous Mortality Investigation reports (Continuous Mortality Investigation, 2015). Being official proposals of the actuarial Societies, such tables enjoy wide use and are also heavily used in the valuation of pension and post-retirement medical liabilities. For example, in the US the SOA tables have been included by the Internal Revenue Service for the purposes of the Pension Protection Act of 2005, or by the Congressional Budget Office for long-term forecasts.

The basic aim in constructing the tables is to convert the raw mortality data into a graduated table of yearly mortality rates and improvement factors, broken down by age and gender. Since the goal is to forecast future mortality from retro-

spective experience, the process involves two fundamental steps: *smoothing* raw data to remove random fluctuations resulting from finite data sizes; and *extrapolating* future rates. To maximize actuarial credibility of the tables, graduation techniques are applied, in particular for estimating mortality improvement trends based on past experience and then projecting those trends into future years. For example, see the RPEC reports SOA (2014b,a) for the full description of constructing the US tables/scales, as well as more general SOA longevity studies in Purushotham et al. (2011); Rosner et al. (2013).

In the present article, we propose a new methodology to graduate mortality rates and generate mortality improvement scales within a single statistical model. More precisely, we advocate the use of Gaussian process regression, a type of Bayesian nonparametric statistical model. Our aim is to provide a data-driven procedure that produces an alternative to existing methods while enjoying a number of important advantages:

- The GP framework is Bayesian, offering rich uncertainty quantification. The model produces mortality curves smoothed over multiple dimensions, as well as credible intervals which quantify the uncertainty of these curves. This is generated for in-sample smoothing and out-of-sample forecasts. In their basic form, the latter forecasts are Gaussian, allowing for a simple interpretation of the uncertainty by the actuary. Moreover, the GP model is able to generate stochastic *trajectories* of future mortality experience. We demonstrate this projection over both age and calendar year, but the GP model can be consistently applied over higher dimensional data as well. From this, full predictive distributions for annuity values, life expectancies,

and other life contingent cash-flows can be produced. Such analyses can provide core components of stress testing and risk management of mortality and longevity exposures.

- Using GPs leads to unified modeling of mortality rates and mortality improvement; one may analytically differentiate the mortality surface to obtain mortality trends (and corresponding credible bands) over calendar years. This structure offers a coherent approach to both tables, jointly quantifying uncertainty in rates and improvement factors.
- Standard graduation techniques are sensitive to edge issues, i.e. the experience in the latest few years. For example, to achieve a better prediction, the MP-2015 method extrapolates rates from 2010 onwards, effectively excluding the last 5 years of data (as of this writing, CDC data go up to 2014). In contrast, our GP approach intrinsically handles the specific shape of the data and is well suited to incorporating missing data. Therefore, dropping the "edge years" is not necessary with GP, with its self-adjusting credible bands.
- The GP approach provides natural "updating" of mortality tables in terms of incorporating the latest mortality experience. The end users can easily update the tables, no longer requiring reliance on official updates.

To recapitulate, the main contribution of this article is to propose the use of Gaussian process regression for constructing mortality tables and improvement factors. While being a relatively new "machine learning" paradigm, the underlying statistical methodology and most crucially the software implementa-

tion has matured significantly in the past decade. To wit, all of the case studies below have been implemented straightforwardly using publicly-available, free, well-documented software, and required only basic programming skills. With a much shorter learning curve and enhanced functionality, the GP approach is well-positioned to be the 21^{st} century framework for mortality data analysis.

From the empirical direction, our data-driven analysis sheds light on the latest mortality experience, whereby mortality improvements appear to have significantly moderated from past trends. Specifically, after implementing the above framework on the latest US mortality experience, we document that as of 2015, mortality improvement factors are (statistically) zero, and possibly *negative* for ages 55–70 since as early as 2012. These estimates diverge significantly from SOA projections embedded in MP-2015 that continue to bake in past improvements. Lower mortality improvement rates would have a material impact across the pension industry. This paper offers statistical support to the anecdotal demographic evidence of declining US longevity and calls into question traditional backward-looking methods for constructing mortality improvement factors.

6.2.1 Comparison to Other Approaches

Mortality experience is summarized by a mortality surface, indexed by Age (rows i) and calendar Year (columns j). Typical data consists of two matrices $\boldsymbol{D}$ and $\boldsymbol{E}$ (or $\boldsymbol{L}$), listing the number of deaths D, exposed-to-risk E, or the mid-year population L, respectively. In the first step, one postulates a relationship between the individual elements of these matrices, D_{ij} and E_{ij}, in terms of the latent (logarithmic) mortality state μ_{ij}. In the second step, one estimates μ_{ij}

through a statistical fitting approach. We may identify two classes of estimation: (i) data-driven models that infer μ_{ij} by statistical smoothing techniques; (ii) factor models that express μ_{ij} in terms of several one-dimensional indices. For example, in Age-Period-Cohort ("APC") models those factors are additive and correspond to Age, Year and Cohort effects; in Lee-Carter (Lee and Carter, 1992) models they correspond to Age, Year, and an Age-Year interaction term. A common distinction is to assume a non-smooth evolution of the mortality surface in time, coupled with a smooth Age effect. The latter Age-modulating terms are then fitted non-parametrically by maximum likelihood, or given a fixed functional form, such as linear or quadratic in Age (Cairns et al., 2006; Hunt and Blake, 2014). Imposing an underlying one-dimensional structure facilitates interpretation of the fitted output, but potentially leads to model risk. In contrast, the data-driven methods, dating back to the classical graduation technique of Whittaker (1922), maintain an agnostic view of mortality experience, and solely focus on removing random fluctuations in observed deaths. Modern frameworks typically work with various types of splines, extending the seminal work by Currie et al. (2004) (see also a modern software implementation in Camarda (2012)). Here, the main challenge is appropriate smoothing across both Age and Year dimensions; some of the proposed solutions include constrained and weighted regression splines (Hyndman and Ullah, 2007b), extensions to handle cohort effects that generate "ridges" (Dokumentov and Hyndman, 2014), and a spatio-temporal kriging approach (Debón et al., 2010). A mixed strategy of first smoothing the data non-parametrically, and then inferring underlying factor structure was proposed and investigated in Hyndman and Ullah (2007b). Finally, we also mention

Bayesian approaches (Czado et al., 2005b; Girosi and King, 2008) that replace MLE-based point estimates with a posterior distribution of the mortality rate. To date, there is no consensus on which framework is the most appropriate. For example, the influential study by Cairns et al. (2009b) considered eight different mortality models. Another recent study by Currie (2016) looked at 32 models, nesting the former eight.

A further reason for the large number of models is the use of different link functions (log-Poisson, logit-Poisson, logit-Binomial, etc.), that connect the logarithmic mortality state to deaths and exposures. These modeling choices are important since they affect the optimization procedure (usually some variant of maximum likelihood) applied in calibrating each model. The Binomial model is defined as $D_{ij} \sim Bin(E_{ij}, e^{\mu_{ij}})$ (Hyndman and Ullah, 2007b); the Poisson model $D_{ij} \sim Poisson(E_{ij}e^{\mu_{ij}})$ (Brouhns et al., 2002b); and the Gaussian model $\frac{D_{ij}}{E_{ij}} \sim \mathcal{N}(e^{\mu_{ij}}, \sigma^2 E_{ij})$ (Girosi and King, 2008). A related issue is regularization of the estimated factors that can be achieved via penalization, see Delwarde et al. (2007); Currie (2013).

In terms of forecasting future mortality, a popular strategy is to differentiate the treatment of the Age index, which is incorporated directly into the mortality state and smoothed appropriately, vis-a-vis the Year index, whose impact is estimated statistically using time-series techniques. This is the basic idea of Lee-Carter models, which construct a time-series process for the Year factor(s) to extrapolate mortality trends and assess forecast uncertainty. More generally, this can be viewed as a principal component approach, expressing the Age-effect as a smooth mortality curve $\mu_t(x_{ag})$, fitted via functional regression or singular value

decomposition techniques, and then describing the evolution of this curve over time (Renshaw and Haberman, 2003; Hyndman and Ullah, 2007b) as a multivariate time-series. In contrast, in the pure smoothing methods, all covariates are given equal footing, and forecasting is done by extrapolating the fitted *surface* to new input locations.

Precise methods for constructing mortality tables are not without controversy, especially when it comes to extreme age longevity or future forecasts. Ideally one ought to just let the "data speak for itself". However, this is in fact a very challenging issue, not least because the question of predictive forecasting must acknowledge that any given fixed forecast is only a *point estimate*, and that there is always an element of uncertainty around the prediction. A common paradigm is to specify a stochastic model for mortality which directly prescribes future uncertainty. This is especially relevant for risk management or pricing applications, where the actuary wishes to incorporate (and hopefully manage) mortality risks. However, most stochastic mortality frameworks are "reduced-form" in the sense of specifying a low-dimensional stochastic system with just a few parameters/degrees-of-freedom. For implementation, one "calibrates" the model to data by minimizing e.g. the mean-squared error. In contrast, the RP-2014 mortality table is bottom-up, aiming to directly specify the full mortality experience with minimal a priori specifications. Relative to these two basic strategies, the approach proposed in this article views uncertainty in forecasts as intrinsic to the statistical model, so that all credible bands are obtained simultaneously both in-sample and out-of-sample.

6.2.2 Mortality Dataset

Our study is US-centric and originated from discussions of the SOA's MP-2014 and successor tables. There has been some controversy that the scale excluded more recent trends, specifically a slowing of mortality improvement that was not fully reflected in the MP-2014 tables. Indeed, a year later, the SOA updated the MP-2014 tables to the MP-2015 tables to include two additional years of mortality experience, and the new tables did in fact reflect a material drop in mortality improvement. In the interim, the CDC has also released new data showing a continued decline in mortality improvement levels.

The mortality data we use comes from Centers for Disease Control (CDC). The CDC data covers ages 0–84 and goes up to 2014 as of the time of writing. For each cell of the table, the CDC data specifies the raw mortality rate for the exposed population. The mid-year exposures L_{ij} are based on inter-censal estimates interpolated based on the 2000 and 2010 census counts. Thus, $e^{\mu_{ij}}$ corresponds to central death rates. Table 6.1 provides a snapshot of the latest year of CDC data (2014). The rapid decrease in sample size causes large variability in reported mortality rates at extreme ages. For a visual representation, two representative years of raw CDC data for Males aged 60–70 are plotted as the solid lines in Figure 6.1. The figure shows the (super-) exponential increase in mortality with respect to age, along with a clear need for data smoothing.

As our training dataset, we used the CDC database covering ages 50–84 in years 1999–2014. (Another data source is provided by Social Security Administration (SSA) and was utilized by RPEC.) Since our main aim is to obtain the *present* mortality rates and to forecast short-term calendar trend through estimat-

Inputs x^n		Log Mortality Rate y^n		Mortality Rate $\exp(y^n)$	
Age (x^n_{ag})	Year (x^n_{yr})	Male	Female	Male	Female
50	2011	-4.931	-5.437	0.00722	0.00435
64	2011	-4.264	-4.707	0.01406	0.00901
74	2011	-3.435	-3.821	0.03222	0.02191
84	2011	-2.408	-2.714	0.08999	0.06625

Table 6.1: Excerpt of CDC mortality data to compare exposures and mortality rates over Ages and gender for calendar year 2011. *Mortality* is the observed proportion D^n/L^n of the deceased during the Year relative to the mid-year population.

ing mortality improvement factors, we only consider older ages and recent years. Our main philosophy is of mortality evolving as a non-stationary surface in Age and Year, so that distant mortality experience is less influential for our analysis. Thus, we purposely leave out (i) young ages which have further features, such as infant/teen mortality, and (ii) most 20^{th} century data. We refer to Li and O'Hare (2015) for a discussion about "local" versus "global" approaches to mortality. To understand the impact of excluding some data, we also considered several subsets listed in Table 6.2.

In comparison to our dataset, the most recent MP-2015 scales incorporate actual smoothed rates up to 2009 with projections thereafter. However, the CDC already provides actual mortality experience up to 2014. The SOA at this stage is still grappling with how to supplement its analyses with the additional 5 years of mortality experience (SOA, 2015).

Set Name	Training Set	Test Set
All Data	1999–2014, ages 50–84	N/A: In-Sample
Subset I	1999–2010, ages 50–84	2011–2014, ages 50–84
Subset II	1999–2010, ages 50–84 & 2011–2014, ages 50–70	2011–2014, ages 71–84
Subset III	1999–2010, ages 50–70	2011–2014, ages 71–84

Table 6.2: Data sets used in analysis. Mortality data is taken from CDC as described in Section 6.2.2.

6.3 Gaussian Process Regression for Mortality Tables

In this paper, we focus on analyzing mortality rates over a two-dimensional input space, namely Age and Year. The mortality data is viewed as a table of N "cells" (see rows of Table 6.1), represented by inputs x^n and outputs or responses y^n, $n = 1, \ldots, N$. In our case, x^n is in fact a tuple and represents the pair (x^n_{ag}, x^n_{yr}). For example, $x^n = (78, 2016)$ is the input for "78-year old in 2016" cell. We use the logarithmic central mortality rate for y^n, namely $y^n = \log(D^n/L^n)$ where D^n and L^n represent the annual deaths and midyear count of lives, respectively, for the n-th cell. The overall inputs $\mathbf{x} = x^{1:N}$ and observations $\mathbf{y} = y^{1:N}$ are denoted by boldface and aggregated into the mortality dataset $\mathcal{D} = (\mathbf{x}, \mathbf{y})$. Superscripts identify individual inputs/outputs, subscripts distinguish coordinates, e.g. x^n_{ag}.

Remark. This point of view treats calendar Year as simply another covariate and is easily extendible to further input dimensions, such as Select Period, etc. Also the format easily allows for missing cells, which, for example, is a common issue for dealing with extreme ages (95+).

6.3.1 Basics of Gaussian Processes

In traditional mortality regression, a parametric function, f, is postulated which maps the inputs $\mathbf{x}$ to the noisy measurements of the log-mortality rate, $\mathbf{y}$. A cell is modeled as

$$y^i = f(x^i) + \epsilon^i, \tag{6.1}$$

where ϵ^i is the error term. With a GP, the function f is deemed to be latent and is modeled as a random variable. Consequently, a GP is defined as a set of random variables $\{f(x)|x \in \mathbb{R}^d\}$ where any finite subset has a multivariate Gaussian distribution with mean $\boldsymbol{m}(\cdot)$ and covariance $\boldsymbol{C}(\cdot,\cdot)$. That is for any n-tuple $\mathbf{x} = x^{1:n}$:

$$f(x^1), \ldots, f(x^n) \sim \mathcal{N}\left(mean = \boldsymbol{m}(\mathbf{x}), covariance = \boldsymbol{C}(\mathbf{x},\mathbf{x})\right).$$

In shorthand, we write $f(\mathbf{x}) \sim GP(\boldsymbol{m}(\mathbf{x}), \boldsymbol{C}(\mathbf{x},\mathbf{x}))$. An important concept of a GP is that each mortality rate is correlated with every other mortality rate: above, $\boldsymbol{C}$ is a $n \times n$ matrix with entry $C(x^i, x^j)$ representing the covariance between the i-th and j-th cells.

Once we collect data $\mathcal{D}$, the next step is to determine the posterior distribution for f, namely $p(f|\mathcal{D})$. That is, we want to know the distribution of mortality rates, given the experience data. Using Bayes' rule, we have

$$p(f|\mathcal{D}) \propto p(\mathbf{y}|f,\mathbf{x})p(f) = \{likelihood\} \cdot \{prior\}$$

where $p(\mathbf{y}|f,\mathbf{x})$ is the "likelihood" and $p(f)$ the "prior". To complete the definition

of the GP, we therefore need to define the "prior", $p(f)$. This is equivalent to setting the initial assumptions for mean function $\mathbf{m}$ and covariance function $\mathbf{C}$.

The Prior Mean Function: the prior mean $m(x)$ stands in for our belief about mortality rate at input x in the absence of any historic data. We might, for example, define $m(\cdot)$ as a Gompertz or Makeham curve in the age coordinate x_{ag}. However, we will show that the choice of $m(\cdot)$ has little impact on the output of the GP model for purposes of *in-sample* smoothing. Even if we set $m(x) = 0$ or $m(x) = \beta_0$ for some constant β_0 and for all x, the results will be largely unaffected, since the posterior mean is largely dominated by the impact of the data. However, for purposes of *out-of-sample* projections, we will conversely show that a more realistic choice of $m(\cdot)$ is required for long term mortality projections.

The Covariance Function: A core concept of a GP is that for any cells i, j, if x^i and x^j are deemed to be "close", then we would expect the outputs, y^i and y^j, to be "close" too. For example, the mortality rate for a 60 year old in 2016 ($x^i = (60, 2016)$) will be closer to that of a 61 year old in 2017 ($x^j = (61, 2017)$), than that of a 20 year old in 1990 ($x^j = (20, 1990)$). This idea is mathematically encapsulated in C: the closer x^i is to x^j, the larger the covariance $C(x^i, x^j)$. It follows, that if x^i and x^j are very close, knowledge of y^j will greatly affect our expectations of y^i. Conversely, if x^i is far from x^j, then y^j will have little influence on our expectations of y^i.

The Posterior Function: To project mortality, we evaluate the GP function on new Age and/or Year inputs $\mathbf{x}_*$, i.e. evaluate $\boldsymbol{f}_* = f(\mathbf{x}_*)|\mathcal{D}$. We show in the next subsection that when m is a constant and the likelihood function is Gaussian, then the posterior distribution for $\boldsymbol{f}_*$ can be determined analytically. In fact, this

posterior itself is a new GP $\boldsymbol{f}_*(\mathbf{x}_*)|\mathcal{D} \sim GP(\boldsymbol{m}_*(\mathbf{x}_*), \boldsymbol{C}_*(\mathbf{x}_*, \mathbf{x}_*))$ with an updated mean and covariance functions, specified in (6.6). The posterior mean $\boldsymbol{m}_*(\mathbf{x}_*)$ is interpreted as the model prediction for inputs $\mathbf{x}_*$, and the posterior covariance $\boldsymbol{C}_*(\mathbf{x}_*, \mathbf{x}_*)$ gives a goodness-of-fit measure for this prediction.

The posterior function can be used for both projecting mortality, as well as producing in-sample smoothed mortality curves. For the latter, all we need to do is set $\mathbf{x}_* = \mathbf{x}$, namely the training set inputs. In this case, the mean $\boldsymbol{m}_*(\mathbf{x})$ of the posterior will produce a smooth set of mortality rates, and the posterior variance $\boldsymbol{C}_*(\mathbf{x}, \mathbf{x})$ quantifies the uncertainty around $\boldsymbol{m}_*(\mathbf{x})$. Alternatively, if $\mathbf{x}_*$ represents inputs of future calendar years, then the posterior will produce an out-of-sample projection of the mortality curves. By fitting a GP, and then analyzing the posterior we are able to achieve the following:

- Estimate the historic smoothed mortality curves by calendar year ($\boldsymbol{m}_*(\mathbf{x})$ above);
- Estimate a credible interval around such curves (use the posterior covariance $\boldsymbol{C}_*(\mathbf{x}, \mathbf{x})$);
- Project the curves forward ($\boldsymbol{m}_*(\mathbf{x}_*)$ for future inputs $\mathbf{x}_*$);
- Estimate the credible intervals for such projections ($\boldsymbol{C}_*(\mathbf{x}_*, \mathbf{x}_*)$);
- Generate stochastic future forecasts (sample from the random vector $\mathbf{f}_*(\mathbf{x}_*)$ as a future mortality scenario);
- Smooth curves over all dimensions, using automatically determined tuning parameters.

Note that the above projections are about $\boldsymbol{f}_*$. An actuary might also wish to

project future mortality experience $\mathbf{y}_*$. In the GP framework, the observations are viewed as the latent f plus noise, so that the marginal credible intervals of $\mathbf{y}_*$ are necessarily wider. When the noise ϵ has a Gaussian distribution, $\mathbf{y}_*$ in fact remains a GP with same mean as $\boldsymbol{f}_*$, and a modified variance due to the variance of ϵ. Practically, forecasting realized mortality (for example, in connection with realized annuity payouts) requires also predicting future exposures E.

Remark. In a Lee-Carter framework one first postulates a parametric form for the mortality experience, such as

$$\mu_{ag,yr} = \alpha_{ag} + \beta_{ag}\kappa_{yr} + \epsilon_{ag,yr} \tag{6.2}$$

where $\boldsymbol{\alpha}$ is the Age shape, $\boldsymbol{\beta}$ is the age-specific pattern of mortality change and $\boldsymbol{\kappa}$ is the Year trend. In the second step, after fitting $\boldsymbol{\alpha}, \boldsymbol{\beta}$ by maximum likelihood, one then postulates a time-series model for the κ factor. Relative to a pure regression model such as ours, the Lee-Carter method treats Age and Year dimensions completely differently; moreover the fit for the Age/Period factors is done globally (i.e. from the full dataset used), so that even spatially distant data directly influences all predictions. Finally, Lee-Carter has no mechanisms for (i) smoothing in-sample experience (beyond model calibration), and (ii) incorporating the uncertainty of the Age/Period factors in out-of-sample forecasts; its forecasts are stochastic only insofar as the time-trend is uncertain. Here we mention that there have been numerous extensions of Lee-Carter, addressing both more complex alternatives to (6.2), as well as other observation settings, such as Poisson-based projections (Brouhns et al., 2002b; Czado et al., 2005b).

6.3.2 Mathematical Details

GP regression takes a response surface approach, postulating an unknown, nonparametric functional dependence between covariates (inputs) $\mathbf{x}$ and outputs $\mathbf{y}$,

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) + \epsilon, \tag{6.3}$$

where f is the *response surface* (or regression map) and ϵ is the mean-zero noise term with observation variance $\sigma^2(x)$, independent across x's. The meaning of the noise term are the statistical fluctuations that lead to deviations between observed raw mortality rates and the latent "true" rates that are being modeled. The strength of these fluctuations $\sigma(x)$ is interpreted as the credibility of the corresponding mortality cell exposure. We remind the reader that throughout the paper, y^n represents log-mortality, and $x^n = (x^n_{ag}, x^n_{yr})$ is a two–dimensional age–year pair. In Gaussian process regression, the map f is assumed to be a realization of a Gaussian process with covariance kernel C that controls the spatial smoothness of the response surface. The GP model starts with a prior on f's over the function space $\mathcal{M}$ and then computes its posterior distribution conditional on the data $\mathcal{D}$. The function space specifying potential f's is a reproducing kernel Hilbert space based on the kernel C. The GP assumption that f is generated by a Gaussian process implies that the posterior distributions are also Gaussian. Hence at any fixed input x, the marginal posterior is $f_*(x) \sim \mathcal{N}(m_*(x), C_*(x,x))$, where m_* is the predictive mean (also the posterior mode, hence maximum a posteriori (MAP) estimator), and $C_*(x,x)$ is the posterior uncertainty of m_*. $C_*(x,x)$ offers a principled empirical estimate of model accuracy, serving as a proxy for the

mean-squared error of m_* at x.

A GP model $GP(\boldsymbol{m}(\mathbf{x}), \boldsymbol{C}(\mathbf{x}, \mathbf{x}))$ is specified through its mean function $m(x^i) = \mathbb{E}[f(x^i)]$ and covariance $C(x^i, x^j) = \mathbb{E}[(f(x^i) - m(x^i))\,(f(x^j) - m(x^j))]$. Specifically, the prior of $\boldsymbol{f}(\mathbf{x})$ is $p(\boldsymbol{f}|\mathbf{x}) = \mathcal{N}(\mathbf{m}, \boldsymbol{C})$, where $\mathbf{m} = (m(x^i))_{1\leq i\leq N}$ and $\boldsymbol{C} = (C(x^i, x^j))_{i,j}$. In the standard case, it is further assumed that the noisy observations vector $\mathbf{y}$ has a Gaussian relationship to the latent $\boldsymbol{f}$, i.e. $\epsilon^i \sim \mathcal{N}\,(0, \sigma^2(x^i))$, so that

$$p(\mathbf{y}|\boldsymbol{f}) = \mathcal{N}(\mathbf{y}|\boldsymbol{f}, \boldsymbol{\Sigma}), \tag{6.4}$$

where $\boldsymbol{\Sigma} = diag(\sigma(x^i)^2)$ is the $N \times N$ noise variance matrix. Certainly, assuming ϵ to be Gaussian with a prescribed variance is not realistic for mortality modeling, but as we show this has minimal statistical effect; we return to this point later. Equation (6.4) implies that if $\boldsymbol{f} \sim GP(\mathbf{m}(\mathbf{x}), \boldsymbol{C}(\mathbf{x}, \mathbf{x}))$ then $\mathbf{y} \sim GP(\mathbf{m}(\mathbf{x}), \boldsymbol{C}(\mathbf{x}, \mathbf{x}) + \boldsymbol{\Sigma})$.

Thanks to the Gaussian assumption, determining the posterior distribution $p(\boldsymbol{f}|\mathbf{y})$ reduces to computing the predictive mean $\boldsymbol{m}_*$ and covariance $\boldsymbol{C}_*$. Combining the above likelihoods and denoting by Θ the hyper-parameters of the GP model, the log-likelihood is

$$\log p(\mathbf{y}|\mathbf{x}, \Theta) = -\frac{1}{2}\mathbf{y}^T(\boldsymbol{C} + \boldsymbol{\Sigma})^{-1}\mathbf{y} - \frac{1}{2}\log|\boldsymbol{C} + \boldsymbol{\Sigma}| - \frac{N}{2}\log(2\pi),$$

where $\mathbf{y}^T$ denotes vector transpose.

The basic GP model treats the prior mean function m as given (i.e. known and fixed). In Section 6.3.3 we discuss the more relevant case where we simultaneously

infer a parametric prior mean $m(x)$ and the kernel hyperparameters, which is known as Universal Kriging. For now, by de-trending via $\mathbf{f} - \boldsymbol{m}(\mathbf{x})$, we may assume without loss of generality that $\boldsymbol{f}$ is centered at zero and $m \equiv 0$. The resulting posterior distribution $\boldsymbol{f}_*(\mathbf{x}_*)$ at a vector of inputs $\mathbf{x}_*$ is multivariate Gaussian (Roustant et al., 2012b) with mean/covariance:

$$\boldsymbol{f}_*(\mathbf{x}_*|\mathbf{x},\mathbf{y}) \sim GP\Big(mean = \boldsymbol{C}(\mathbf{x},\mathbf{x}_*)^T(\boldsymbol{C}+\boldsymbol{\Sigma})^{-1}\mathbf{y}, \tag{6.5}$$

$$covariance = \boldsymbol{C}(\mathbf{x}_*,\mathbf{x}_*) - \boldsymbol{C}(\mathbf{x},\mathbf{x}_*)^T(\boldsymbol{C}+\boldsymbol{\Sigma})^{-1}\boldsymbol{C}(\mathbf{x}_*,\mathbf{x})\Big), \tag{6.6}$$

where $\boldsymbol{C}^T$ is the transpose of $\boldsymbol{C}$.

The effect of (6.6) is that if we have new inputs $\mathbf{x}_*$, then draws from the posterior distribution of $\boldsymbol{f}_*$ at $\mathbf{x}_*$ will be primarily influenced by historic data that have inputs close to $\mathbf{x}_*$. Marginally at a single cell x_*, and similar to kernel regression, the predicted value $m_*(x_*)$ is a linear combination of observed y^i's, capturing the idea of the GP model nonparametrically smoothing the raw mortality data. The covariance kernel C quantifies the relative contribution of different y^i's in terms of the distance of their x^i's to x_*, see Section 6.3.3 below. The observation noise matrix $\boldsymbol{\Sigma}$ represents the credibility of the corresponding observations y's and is used by the GP to automatically determine how much of interpolation versus smoothing to carry out; in the limiting case $\sigma = 0$, the posterior mean exactly interpolates the observation y^i at x^i: $m_*(x^i) = y^i$.

In reality, the credibility of mortality experience is non-constant because of the different number of exposed-to-risk in different age brackets. This suggests that realistically $\sigma(x^i)$ is non-constant, which would in principle influence the con-

tribution of respective cells to the GP predictions. The mortality table structure can be used to estimate $\sigma(x^n)$: $L^n \cdot \exp\{y^n\}$ is expected to be binomially distributed with parameters $p^n \doteq D^n/E^n$, and size $E^n \simeq L^n + D^n/2$. We then have $\mathrm{Var}\left(\exp\{y^n\}\right) = p^n(1-p^n)/E^n$, and large population L^n implies the delta-method estimate

$$\sigma^2(x^n) = \mathrm{Var}\left(y^n\right) \simeq \frac{(1-p^n)}{p^n E^n}. \tag{6.7}$$

It is well known (see e.g. Currie et al. (2004)) that mortality data exhibit *over-dispersion* relative to (6.7), partly due to the fact that the computed p^n is not the true mortality rate. Consequently, some care must be taken with regards to parameter uncertainty when using (6.7). In Currie et al. (2004) and within the context of a Poisson GLM model, this was adjusted by fitting an age-dependent overdispersion factor with a spline. However, unlike a GLM where a correct representation of $\sigma(x^n)$ is crucial due to its interdependence with the link function, its use in GP is only for noise smoothing, so only a reasonable estimate is required. Specifically, in our main analysis we take $\sigma^2(x^n) \equiv \sigma^2$ to be an unknown constant, estimated as part of fitting the model. We return to this issue in Section 6.4.

6.3.3 Covariance Kernels and Parameter Estimation

Given the covariance kernel C, (6.6) fully specifies the posterior distribution $\boldsymbol{f}_*(\mathbf{x}_*)|\mathcal{D}$ conditional on the dataset $\mathcal{D}$. GP inference is thus reduced to simply applying the above formulas, akin to the ordinary least-squares (OLS) equations that specify the coefficients of a linear regression model. Of course in practice the kernel C is not known and must be inferred itself. This corresponds to fitting the

hyperparameters Θ.

Our examples use the separable, spatially-stationary kernel of the squared-exponential family, which written out explicitly takes

$$C(x^i, x^j) = \eta^2 \exp\left(-\frac{(x^i_{ag} - x^j_{ag})^2}{2\theta^2_{ag}} - \frac{(x^i_{yr} - x^j_{yr})^2}{2\theta^2_{yr}}\right). \tag{6.8}$$

In (6.8), covariance between y^i and y^j is determined by the distance between inputs of the respective cells, measured through the (squared) difference in Ages and Years between x^i, x^j, and modulated by the θ's. This use of spatial dependence can be straightforwardly extended to incorporate other dimensions, such as year-of-birth cohorts to conduct an APC allocation, or to include duration, to create a select and ultimate mortality table in the context of life insurance mortality analysis.

The hyper-parameters θ_ℓ are called characteristic length-scales and their effect on the model is quite subtle. Informally, larger θ's result in smoother mortality curves, i.e. correlation dissipates slower. Smaller lengthscales reduce smoothing and lead to "rougher" curves. (The form of (6.8) implies that the mortality curves are infinitely differentiable both in Age and Year dimensions.) Note that the two lengthscale parameters θ_ℓ for Age and Year are different, so that the covariance kernel is anisotropic. The lengthscales also determine the speed at which the latent process reverts back to its prior outside the dataset. For example, considering the Year coordinate and the question of projecting mortality rates into the future, the GP prediction will automatically blend smoothed mortality rates derived from the experience data and the specified Year trend. Indeed, $\boldsymbol{m}_*(\mathbf{x}_*)$ is a weighted

average of observed experience $\mathbf{y}$, and $\mathbf{m}(\mathbf{x}_*)$, with the weights determined by the lengthscale parameters θ_{yr} and θ_{ag}. We contrast this to APC-type models where such blending is *ad hoc* based on user-defined parameters.

Two further GP parameters are the process variance η^2 which controls the natural amplitude of f and the observation noise σ^2 in (6.1) which is viewed as a constant to be estimated. Thus, the overall hyperparameter set is $\Theta \doteq (\theta_{ag}, \theta_{yr}, \eta^2, \sigma^2)$.

The classical method for inferring Θ is obtained by optimizing the marginal likelihood $p(\mathbf{y}|\mathbf{x}, \Theta) = \int p(\mathbf{y}|\boldsymbol{f}, \boldsymbol{\Theta}) p(\boldsymbol{f}|\mathbf{x}, \boldsymbol{\Theta}) d\boldsymbol{f}$ which can be written out explicitly since all the integrands are Gaussian. This leads to a nonlinear optimization problem of simultaneously fitting θ_ℓ's and variance terms η^2, σ^2. Details on this procedure can be found in Section 3.2 of Picheny and Ginsbourger (2013). Alternatively, it is possible to directly specify C, for example from expert knowledge regarding the expected correlation in mortality rates. Given θ's, the MLEs for η and σ can be analytically inferred (Picheny and Ginsbourger, 2013). This approach increases interpretability of the final smoothing/prediction and makes the GP model less of a black-box.

Fitting the Mean Function

A generalized version of (6.3) incorporates a parametric prior mean of the form $m(x) = \beta_0 + \sum_{j=1}^{p} \beta_j h_j(x)$, where β_j are constants to be estimated, and $h_j(\cdot)$ are given basis functions. The coefficient vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is obtained in parallel with computing $\mathbf{m}_*, \boldsymbol{\Sigma}$. Letting $\boldsymbol{h}(x) \doteq (h_1(x), \ldots, h_p(x))$ and $\boldsymbol{H} \doteq \left(\boldsymbol{h}(x^1), \ldots, \boldsymbol{h}(x^N)\right)$, the posterior mean and variance at cell x are (Roustant et al.,

2012b)

$$\begin{cases} \hat{\boldsymbol{\beta}} = \left(\boldsymbol{H}^T(\boldsymbol{C}+\boldsymbol{\Sigma})^{-1}\boldsymbol{H}\right)^{-1}\boldsymbol{H}^T(\boldsymbol{C}+\boldsymbol{\Sigma})^{-1}\mathbf{y}; \\ m_*(x_*) = \boldsymbol{h}(x_*)\hat{\boldsymbol{\beta}} + \boldsymbol{c}(x_*)^T(\boldsymbol{C}+\boldsymbol{\Sigma})^{-1}(\mathbf{y}-\boldsymbol{H}\hat{\boldsymbol{\beta}}); \\ s_*^2(x_*) = C(x_*,x_*) + \left(\boldsymbol{h}(x_*)^T - \boldsymbol{c}(x_*)^T(\boldsymbol{C}+\boldsymbol{\Sigma})^{-1}\boldsymbol{H}\right)^T\left(\boldsymbol{H}^T(\boldsymbol{C}+\boldsymbol{\Sigma})^{-1}\boldsymbol{H}\right)^{-1}\cdot \\ \qquad\qquad \cdot\left(\boldsymbol{h}(x_*)^T - \boldsymbol{c}(x_*)^T(\boldsymbol{C}+\boldsymbol{\Sigma})^{-1}\boldsymbol{H}\right), \end{cases} \tag{6.9}$$

where $\boldsymbol{c}(x_*) = (C(x_*, x^i))_{1\le i\le N}$. Note that (6.9) reduces to (6.6) when $\boldsymbol{h} \equiv \mathbf{0}$. We also see that the fitted coefficients $\boldsymbol{\beta}$ are in analogue to the classical least-squares linear model. A non-constant mean function is important for imposing structural constraints about the shape of the mortality curve, as well as the long-term improvement trends in mortality rates. Appropriate choices for parameterizing m are needed to be able to give reasonable out-of-sample projections, which corresponds to extrapolating in Age, or in calendar Year.

Use of a mean function for the GP via (6.9) combines the idea of parametrically de-trending the raw data through a fitted Age shape, and then modeling the residual fluctuations into a single step. We note that as m is assigned more and more structure, the residuals necessarily decrease and becomes less correlated. This calls to attention the typical over-fitting concern. Standard techniques, such as cross-validation or information criteria could be applied as safeguards, but their precise performance within the GP framework is not yet fully analyzed. We therefore confine ourselves to a qualitative comparison regarding the impact of the prior mean $m(\cdot)$ on the GP model output.

Bayesian GP and MCMC

One can also consider a fully Bayesian GP model, where the mean and/or covariance parameters have a prior distribution, see Williams and Rasmussen (2006). Bayesian GP implies that there is additional, intrinsic uncertainty about C which is propagated through to the predictive distributions f_*. Starting from the hyper-prior $p(\Theta)$, the posterior distribution of the hyperparameters is obtained via $p(\Theta|\mathcal{D}) \propto p(\Theta)p(\mathbf{y}|\mathbf{x}, \Theta)$. This hierarchical posterior distribution is typically not a GP itself. Practically this means that one draws realizations Θ^m, $m = 1, 2, \ldots$ from the posterior hyperparameters and then applies (6.6) to each draw to compute $m_*(\mathbf{x}_*|\Theta^m), C_*(\mathbf{x}_*, \mathbf{x}_*)|\Theta^m)$.

In general, sampling from $p(\Theta|\mathcal{D})$ requires approximate techniques such as Markov Chain Monte Carlo. The output of MCMC is a sequence $\Theta^1, \Theta^2, \ldots, \Theta^M$ of Θ values which can be used as an empirical approximation for the marginal distribution of Θ, namely $p(\Theta|\mathbf{y}, \mathbf{x})$. From this sequence, it possible to calculate means and modes of the model parameters or use the Θ sequence directly to conduct posterior predictive inference. A hybrid approach first specifies hyperparameter priors but then simply uses the MAP estimates of Θ for prediction (thus bypassing the computationally intensive MCMC steps). This idea is motivated by the observation that under a vague prior $p(\Theta) \propto 1$, the posterior of Θ is proportional to the likelihood, so that the MAP estimator $\hat{\Theta}$ which optimizes $p(\Theta|\mathbf{y}, \mathbf{x})$ becomes identical to the MLE maximizer above.

We note that standard MCMC techniques are not well suited for GP as the components of Θ tend to be highly correlated resulting in slow convergence of the MCMC chains. One solution is to use Hamiltonian Monte Carlo (HMC) (Brooks

et al., 2011) which is better equipped for managing correlated parameters.

Setting Priors for the Bayesian Model

To improve the efficiency of the MCMC routines, we first standardize the input covariates, for example $x^i_{ag,std} \doteq (x^i_{ag} - mean(\mathbf{x}_{ag}))/sd(\mathbf{x}_{ag})$. We then set priors relative to this standardized data model. Note that for comparative purposes with non-Bayesian models, the resulting posteriors of $\boldsymbol{\beta}$ and Θ then need to be transformed back to the original scale.

Priors are taken to be weakly informative, accounting for the specifics of each hyperparameter. For the lengthscale, θ_ℓ should be below the scale of the input x_ℓ, otherwise the resultant model will be essentially linear in the ℓ^{th} input dimension (Carpenter et al., 2016). Thus a prior that curtails values much beyond the data scale is appropriate. After standardization, we found that $\log \theta_\ell \sim \mathcal{N}(0, 1)$ is reasonable. The η parameter plays a role similar to that of the prior variance for linear model weights in a standard linear regression, and we found $\log \eta^2 \sim \mathcal{N}(0, 1)$ prior to be reasonable for the linear and quadratic-mean models. The prior for σ should reflect the noise in the data. For the CDC data, we set the prior $\sigma^2 \sim \mathcal{N}_+(0, 0.2)$, restricted to be positive. When including trend, priors for the β parameters are also required. These are set similarly to standard regression coefficients. In our analysis, we tested both Cauchy priors of Cauchy$(0, 5)$ or Gaussian priors of $\mathcal{N}(0, 5)$ and found both to be reasonable. For the intercept coefficient we chose $\beta_0 \sim \mathcal{N}(-4, 5)$ to reflect log-mortality, whereby $\exp(y) \simeq 2\% = \exp(-3.9)$.

Remark. The Bayesian hierarchical approach for determining the parameters of

the covariance matrix is also coined "automatic relevance determination". The Bayesian model will automatically select the values of θ_ℓ and η without the need for using cross-validation or other approaches to set the parameter levels. Smaller values of θ amplify the effect of the difference calculation in the covariance matrix, hence determining the relevance of an input dimension. Thus the Bayesian approach automatically sets the level of covariance among the y-values.

6.3.4 Software

There are several software suites that implement Gaussian process modeling and can be used for our application. The software is complementary in terms of its capabilities and approaches, in particular for inferring the covariance kernel C and for handling extensions of GPs discussed in Section 6.5 below.

To implement Bayesian GP models, we built models in Stan (Carpenter et al., 2016). Stan is a probabilistic programming language and is a descendant of other Bayesian programming languages such as BUGS and JAGS. In its default setting, Stan's engine utilizes Markov chain Monte Carlo techniques, and in particular a version of Hamiltonian Monte Carlo (HMC) (Brooks et al., 2011). Stan also allows the option of working with the MAP estimate $\hat{\Theta}$ or the incorporation of non-conjugate priors, and implementation of idiosyncratic features within a model. Stan automatically infers the GP hyperparameters, specifically the lengthscales θ's, that determine the smoothness of the mortality curves. This allows for a more data-driven approach compared to traditional graduation that a priori imposes the degree of smoothing to apply to raw data.

Within the `R` environment, we utilized the package "DiceKriging" (Roustant

et al., 2012b). `DiceKriging` can fit both standard and parametric trend (6.9) models, and implements five different kernel families (Gaussian, exponential, Matérn). Moreover, `DiceKriging` can handle non-constant observation noise and has multiple options regarding the underlying nonlinear optimization setup. It estimates hyper-parameters through maximum likelihood (but does not do MCMC).

6.4 Results

We implemented a GP model for CDC mortality rates using a squared-exponential (6.8) covariance structure. To analyze and compare the different choices available within the GP framework we have experimented with:

1. Other covariance kernel families, in particular Matern-5/2;
2. MLE and Bayesian approaches to inference of hyperparameters Θ;
3. A variety of mean function specifications;
4. Choice of inhomogeneous noise variance $\sigma(x)$.

For easier reading, the Figures and Tables below show the results for the Males; most of the conclusions are identical for both genders; where appropriate we make further remarks. Figures and Tables for female data can be found in Appendix 6.7.

We tested both the `DiceKriging` and Stan models as described in Section 6.3.4. Table 6.3 reports the MLE and MAP hyperparameter estimates for the intercept-only models fitted with All data (Males aged 50–84, years 1999–2014, see Table 6.2). All of the MLEs are quite close to the MAP estimates and both

fall in the 80% credible intervals for the MCMC runs. Closer analysis of the Stan output revealed that the hyper-parameter posteriors are reasonably uncorrelated, justifying the use of the MAP estimates and corresponding marginal credible intervals.

Comparing both methods showed the resulting posterior distributions for the GP to be near identical, with the posterior means m_* on average within 0.3% (relative error) of each other, and the credible bands within 1.2% of each other. This indicates stability of the GP estimates given slightly different hyper-parameters.

Consequently, the rest of the analysis in this paper is done using the simpler `DiceKriging` model which is quicker to fit and produces a convenient Gaussian posterior for the log-mortality (the fully-Bayesian model built in Stan can be viewed as a mixture-of-Gaussians). Similarly, there was no major difference in prediction and smoothing when picking different covariance kernels. In general, picking a kernel is like picking a basis family for linear regression; basic caveats apply, but it is mostly a secondary effect. Below we focus on the squared-exponential kernel. One benefit of this choice is that the resulting scenarios f_* are guaranteed to be infinitely differentiable, which enables analytic treatment of instantaneous mortality improvement $\partial_{yr} f_*$, see Section 6.4.4.

For the observation noise, estimating a constant noise variance led to MLE of $\hat{\sigma}^2 = 2.808 \cdot 10^{-4}$. We observe that both the Gaussian assumption and the i.i.d assumptions on the resulting residuals are statistically plausible (cf. Figure 6.14 in Section 6.8.1.) As a further check, we tried to work with a non-constant $\sigma^2(x)$ by plugging-in the delta method estimate in (6.7). However, this led to credible

	`DiceKriging`	Stan		
	MLE	MAP	MCMC Mean	MCMC 80% Posterior CI
θ_{ag}	15.8384	14.9320	10.3580	(4.8976, 16.3939)
θ_{yr}	15.5308	14.4895	24.6674	(12.8976, 38.2304)
η^2	1.8468	1.2372	1.8862	(0.7618, 3.5324)
σ^2	2.808e-04	2.752e-04	2.745e-04	(2.5031e-04, 2.988e-04)
β_0	-3.8710	-3.8277	-3.7966	(-4.5986, -3.0185)

Table 6.3: Hyperparameter estimates based on maximum likelihood (`DiceKriging`) and maximum a posteriori probability (Stan), along with MCMC summary statistics. The GP is fitted to all data and uses squared–exponential covariance kernel (6.8) with prior mean $m(x) = \beta_0$. Stan hyper-priors (on standardized data) were $\log\theta_{ag}, \log\theta_{yr}, \log\eta^2 \sim \mathcal{N}(0,1)$ i.i.d., $\sigma^2 \sim \mathcal{N}_+(0, 0.2)$, $\beta_0 \sim \mathcal{N}(-4, 5)$.

bands that are too narrow in terms of coverage ratios due to the aforementioned over-dispersion effect. Manual calibration found that $\check{\sigma}^2(x) = 4 \cdot (1 - p_x)/(p_x E_x)$, i.e. an overdispersion factor of 2, works fine. The resulting estimated $\check{\sigma}^2$ values ranged over $[1.066 \cdot 10^{-4}, 1.304 \cdot 10^{-3}]$ with a mean of $4.36 \cdot 10^{-4}$. This is close to the constant-σ^2 MLE estimate and the respective projections were very close, confirming that with a GP model the whole question of capturing observation errors is a "higher order" concern. For ease of interpretation, we thus used a constant σ^2, estimated via MLE, for the remainder of the analysis.

6.4.1 Retrospective Analysis

We begin with a retrospective look at smoothed mortality experience over the recent past. Traditionally, this is done using actuarial graduation techniques; for the GP framework smoothing is simply the in-sample prediction $m_*(\mathbf{x})$. Specifically, we fit a model using all the data, and investigate the mortality during the last 5 years of the period. Figure 6.1 shows the estimated mortality rates as a

function of age, specifically Males aged 60–70. The left panel compares the raw and GP-smoothed rates for 2010 and 2014, while the right panel shows the overall yearly trend for years 2010–2014. As a complement to above, Figure 6.2 provides a preliminary analysis of mortality improvement by plotting mortality rates against time. We show the observed and smoothed mortality rates against calendar years 1999–2014 for Males and Females aged 60, 70, and 84, along with the forecasted rates up to 2016. From the figure, we clearly observe the decrease of mortality at older ages which is, however, slowing down in the last few years.

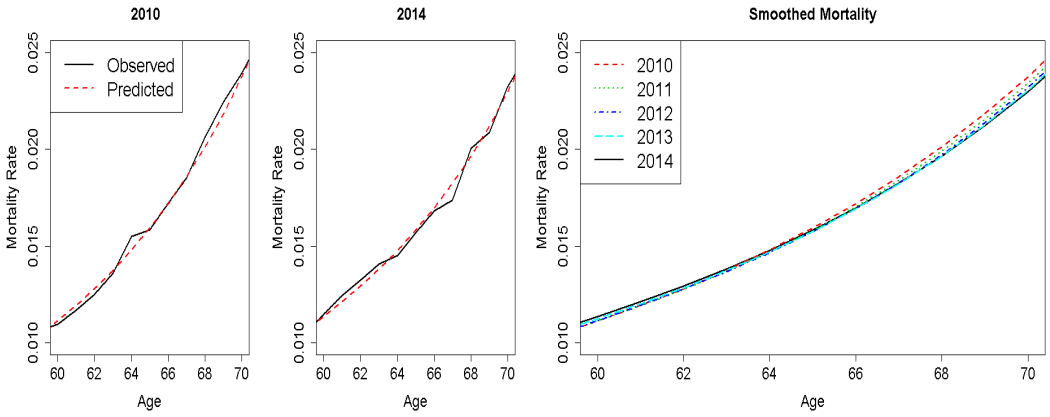

Figure 6.1: Mortality rates for Males aged 60–70 during the years 2010–2014. Raw (solid) vs. smoothed (dashed) mortality curves. Models are fit to 1999–2014 CDC data for Ages 50–84 (All data). Mean function $m(x)$ is intercept-only, $m(x) = \beta_0$.

A key output of official tables are the mortality improvement scales, such as the MP-2015 rates $MI^{MP}_{back}(x_{ag}, yr)$, where we distinguish the common indexing by Age, keeping Year fixed. These are intuitively the smoothed version of the raw annual percentage mortality improvement which is empirically observed via

$$MI^{obs}_{back}\left(x_{ag}; yr\right) \doteq 1 - \frac{\exp\left(\mu(x_{ag}, yr)\right)}{\exp\left(\mu(x_{ag}, yr-1)\right)} \tag{6.10}$$

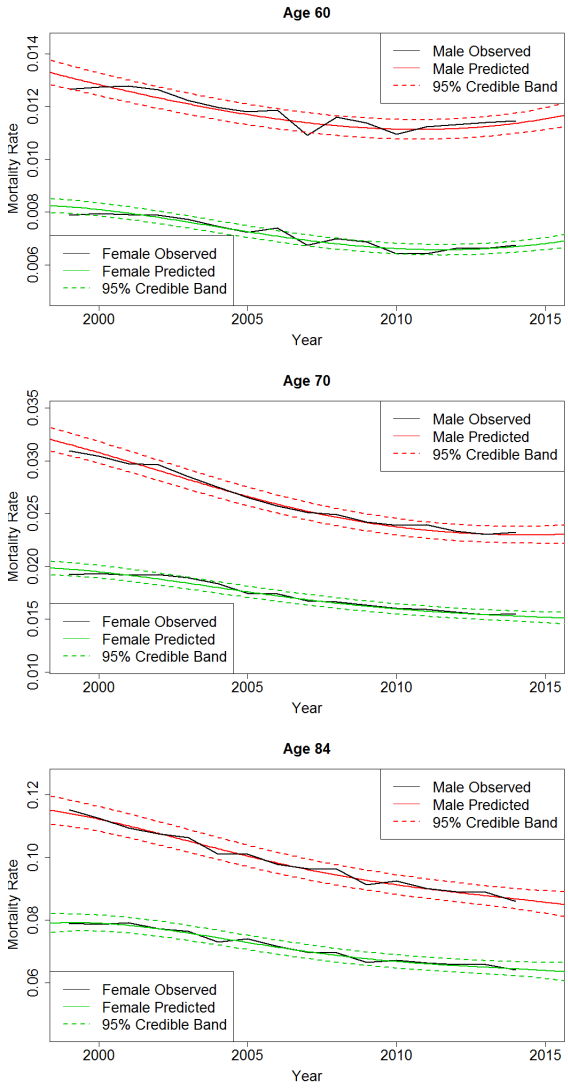

Figure 6.2: Mortality rates for Males (top) and Females (bottom) aged 60, 70 and 84 over time. The plots show raw mortality rates (solid black) for years 1999–2014, as well as predicted mean of the smoothed mortality surface (solid red) and its 95% credible band, for 1999–2016. Models are fit to the 1999–2014 CDC data for Ages 50–84 (All data). Mean function is intercept-only, $m(x) = \beta_0$.

with $\mu(x_{ag}, yr)$ the raw log-mortality rate for (x_{ag}, yr). In analogue to above, we can obtain the predicted mean improvement by replacing μ's by the GP model

posteriors f_*'s and integrating over their posterior distributions:

$$m^{GP}_{back}(x_{ag}, yr) \doteq \mathbb{E}\left[MI^{GP}_{back}(x_{ag}, yr)\right] \doteq \mathbb{E}\left[1 - \frac{\exp\left(f_*(x_{ag}, yr)\right)}{\exp\left(f_*(x_{ag}, yr-1)\right)}\right]. \quad (6.11)$$

Figure 6.3 shows these different improvement scales for ages 50–85 and two sample years, 2000 and 2014; the MP-2015 curves are from the published SOA reports (SOA, 2015). We observe that the raw mortality improvements are extremely noisy, which is not surprising since they are based on the relative difference of two very similar raw mortality rates. Figure 6.3 also indicates that the MP-2015 estimates are significantly higher than either the actual experience (which has moderated a lot in the past decade) or our fit m^{GP}_{back}, with differences of as much as 2% p/a in improvement factors. Figure 6.4 emphasizes that there is a downward trend in mortality improvement, and moreover non-uniform behavior across ages. This throws into question the MP-2015 concept of a sustained, age-uniform projected long-term mortality improvement trend.

6.4.2 Mean Function Modeling

We tested three models for the prior mean function: an intercept-only model $m(x^n) = \beta_0$, a linear model, $m(x^n) = \beta_0 + \beta_1^{ag} x^n_{ag} + \beta_1^{yr} x^n_{yr}$, and a quadratic age model, $m(x^n) = \beta_0 + \beta_1^{ag} x^n_{ag} + \beta_1^{yr} x^n_{yr} + \beta_2^{ag} (x^n_{ag})^2$. Thus, the linear model has the log mortality increasing linearly in age and decreasing linearly in calendar year. The quadratic model then adds a convexity component to the age axis.

The coefficients of these functions were estimated concurrently with fitting the GP models using (6.9). A summary of the models and the coefficient estimates

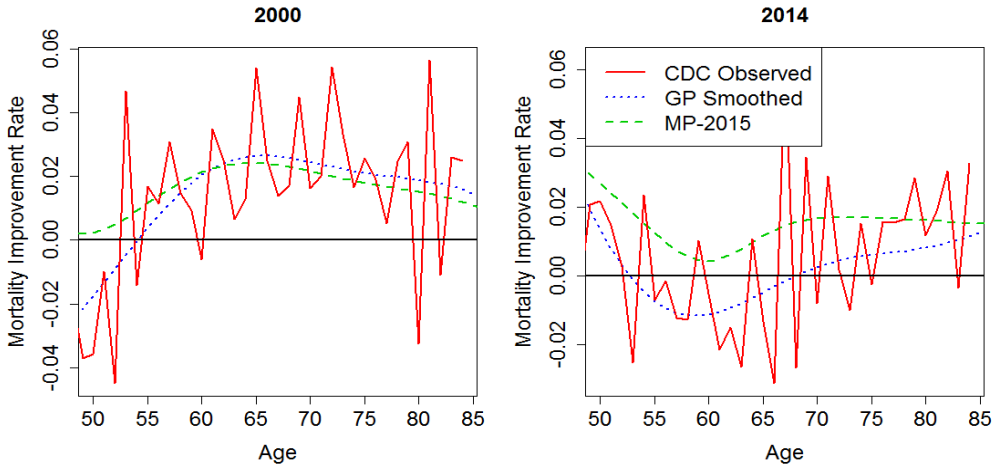

Figure 6.3: Mortality improvement factors for Males using All Data. Solid lines indicate the empirical mortality experience $MI^{obs}_{back}(\cdot; yr)$ for years $yr \in \{2000, 2014\}$, the dotted and dashed lines are $m^{GP}_{back}(\cdot; yr)$ from (6.10), and the MP-2015 improvement scale $MI^{MP}_{back}(\cdot; yr)$, respectively.

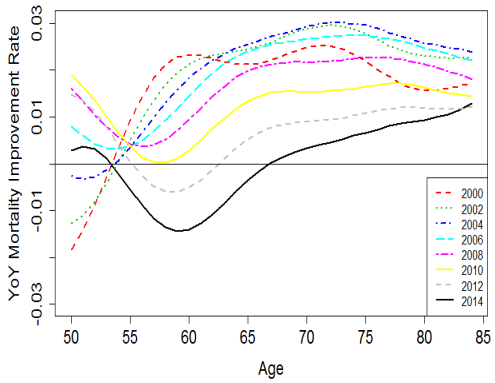

Figure 6.4: Comparison of smoothed yearly mortality improvement factors $m^{GP}_{back}(x_{ag}; yr)$ from (6.10) for Males using All data and $yr = 2000, \ldots, 2014$. The curves for 2000 and 2014 are the same as in Figure 6.3.

is shown in Table 6.4. One finding is that the fitted year-trend coefficient $\beta_1^{(yr)}$ is consistently estimated by both the linear and quadratic model and indicates a linear improvement in log mortality rates of about 1.4% per calendar year in both of these models regardless of assumptions on age shapes. Since this model is fitted

to ages 50–70, these results are consistent with the long-term trend of improving mortality. As expected, the table also indicates a strong Age effect; we note that the fitted coefficient $\beta_2^{(ag)} = 1.459 \cdot 10^{-4}$ for the quadratic age component confirms a significant convexity of log-mortality in Age.

	Mean Function Parameter MLE's			
	β_0	β_1^{ag}	β_2^{ag}	β_1^{yr}
Intercept	-4.526	-	-	-
Linear	18.737	0.081	-	-1.397e-02
Quadratic	19.641	0.064	1.459e-04	-1.417e-02
	GP Hyperparameter MLE's			
	η^2	σ^2	θ_{ag}	θ_{yr}
Intercept	6.213e-01	3.428e-04	8.384	12.746
Linear	8.521e-04	1.761e-04	3.610	3.543
Quadratic	1.403e-03	2.998e-04	3.629	3.475

Table 6.4: Fitted mean function and covariance parameters using Subset III (ages 50–70 and years 1999–2009) for Males. The mean functions are $m(x^n) = \beta_0$ for Intercept, $m(x^n) = \beta_0 + \beta_1^{ag} x_{ag}^n + \beta_1^{yr} x_{yr}^n$ for Linear, and $m(x^n) = \beta_0 + \beta_1^{ag} x_{ag}^n + \beta_1^{yr} x_{yr}^n + \beta_2^{ag} (x_{ag}^n)^2$ for Quadratic.

Intuitively, the mean function provides a fundamental explanation of mortality rates by age and year, while the covariance structure captures deviations from this postulated relationship based on nearby observed experience (with the influence depending on the lengthscale). Consequently, the choice of the mean function affects the covariance structure; a stronger trend/shape lowers the spatial dependence of the residuals. We observe this effect in Table 6.4, where the intercept-only model has length-scales of $\theta_{ag} \approx 8.5, \theta_{yr} \approx 12.5$, while for the linear and quadratic models the range of the length-scales is much smaller $\theta_\ell \approx 3.5$. Another effect of the mean function is on the hyperparameter η which can be viewed as the variance of the model residuals. If the mean function fits well then

we expect smaller η. In turn, smaller η translates into tighter credible intervals around in-sample smoothing and out-of-sample forecasts. Table 6.4 shows that the values for η and σ are similar across linear and quadratic models while the intercept-only model has uniformly larger values across parameters.

Figure 6.5 illustrates these three models fit to Subset III which emulates deep out-of-sample extrapolation. As discussed, out-of-sample forecasting by the GP model can be viewed as blending the data-driven prediction with the estimated trend encapsulated by m. Specifically, as $x^n_{*,ag}$ moves beyond the age range of $\{50, 51, \ldots, 70\}$ in Subset III we have $m_*(x_*) \to m(x_*)$. In the case of an intercept-only model, this implies that $m_*(x_*) \to \beta_0$, i.e. the projected mortality is independent of either Age or Year. In Figure 6.5 the asymptotic projected rate was $\exp(\hat{\beta}_0) = 1.214\%$. A similar issue pertains to the linear-mean model whose long-range forecasts imply exponential Age dependence which is not appropriate for ages above 80. This discrepancy is successfully resolved by the quadratic $m(x_*)$ model. The lengthscales θ control this transition; roughly speaking extrapolating more than θ distance away reduces to $m_*(x_*) \simeq m(x_*)$. This can also be seen in Figure 6.5: since the training data includes up to 2010, information is borrowed much more from past data in the case of 2011, as opposed to 2014. As a result, for the intercept-only model with $\theta_{yr} = 11.461$, the forecast is acceptable in 2011 (as it is driven by trained data up to 2010), but deteriorates dramatically for 2014. This effect is also present but less apparent in the trend models due to smaller values of θ_{yr}, so that the forecasts of the latter models rely more heavily on their mean functions to explain mortality through age and year.

Another way for model comparison is to look at the widths of the respective

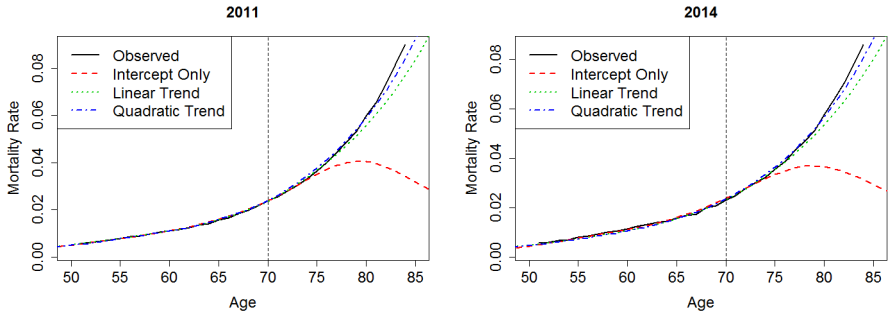

Figure 6.5: Comparison of mean function choices in extrapolating mortality rates at old ages. Models are fit to years 1999–2010 and ages 50–70 (Subset III) for Males, with estimates made for Age 50–85 in 2011 and 2014. The vertical line indicates the boundary of the training dataset in x_{ag}. The mean functions are given in Table 6.4.

credible intervals. For example, for year 2010 age 84, the observed mortality rate was 8.999% and the intercept, linear and quadratic models generated 95% credible intervals of $(0.783\%, 10.254\%)$,$(7.100\%, 8.562\%)$ and $(6.379\%, 11.188\%)$ respectively. Certainly the first interval is too wide (partly due to the large η and θ's of the intercept-only model), while the second interval is too narrow and does not even contain the raw data point (the linear model apparently underestimates η). On the other hand, for age 71 in year 2014, the raw rate was 2.489% and the respective 95% credible intervals were $(2.258\%, 2.927\%)$, $(2.378\%, 2.703\%)$ and $(2.346\%, 2.798\%)$. While all models now contain the observed rate, the linear model again has the tightest credible interval, which might indicate poor goodness-of-fit.

Returning to in-sample smoothing and looking again at Figure 6.5, we observe that all three models generate very similar forecasts for ages 55–70 This confirms that in-sample m_* is data-driven and the choice of $m(\cdot)$ is secondary. To summarize, the most important criterion in including a mean function is whether

the goal is to predict out-of-sample and if so, how far out-of-sample. In general, mean modeling is crucial, but the precise choice of the mean function is not as clear. In Section 6.5.1 we discuss one further method for mean-modeling based on Age-grouping.

6.4.3 Predictive Accuracy

Figure 6.5 can also be viewed as a first glimpse into the predictive accuracy of a GP mortality model. Recall that in the Figure we fit to mortality data from 1999-2010 and then forecast 1 year out (2011) and 4 years out (2014). The Figure then compares these projections to the observed mortality experience in 2011 and 2014. As discussed, these projections are highly sensitive to the choice of $m(x)$, especially in terms of the Age-structure because the models are only given experience up to Age 70 and hence have zero information about high-age mortality.

To provide a more "fair" comparison, Table 6.5 shows projections for other input datasets. Overall, we observe excellent predictive power for making projections 4-years out (fit using Subset I, forecast for 2014), confirming the competitive performance of the GP fitted models.

Beyond the predictive mean $m_*(x_*)$, we also report the corresponding posterior marginal variance $s_*^2(x_*) \doteq C_*(x_*, x_*)$ which is a proxy for the confidence the model assigns to its own prediction. This provides a valuable insight: for example if fitted to ages 50–70 (Subset III) and projecting for age 80 in year 2014: $\tilde{x} \doteq (x_{ag}, x_{yr}) = (80, 2014)$, the intercept-only model reports minimal predictive power which is reflected in the very large $s_*^{(III)}(\tilde{x}) = 0.4565$, in light of which the poor

prediction $m_*^{(III)}(\tilde{x}) = -3.7177$ is simply a "shot in the dark". Indeed, the model predicts mortality rate of 2.43% which is nowhere the realized 5.78%, but is still within its 95% credible interval of $(0.98\%, 6.03\%)$. Including more ages (Subset I) gives a more reasonable and much more confident forecast of $m_*^i(\tilde{x}) = -2.8416$ and $s_*^i(\tilde{x}) = 0.0463$, and including more years (which makes $\tilde{x}$ to be right at the edge of observed data) raises credibility even further, $m_*^{(All)}(\tilde{x}) = -2.8579$ and $s_*^{(All)}(\tilde{x}) = 0.0170$. Table 6.5 also quantifies the gains from using a more sophisticated m – the quadratic trend allows to shrink $s_*^i(\tilde{x})$ from 0.0463 to 0.0333, and brings the prediction $m_*^i(\tilde{x})$ closer to the eventually realized experience.

Intercept-only $m(x) = \beta_0$							
	Fit to Subset III		Fit to Subset I		Fit to All Data		Observed
x_{ag}	$m_*^{(III)}$	$(s_*^{(III)})$	m_*^i	(s_*^i)	$m_*^{(All)}$	$(s_*^{(All)})$	μ
70	-3.7520	(0.0580)	-3.7380	(0.0427)	-3.7702	(0.0169)	-3.7630
80	-3.7177	(0.4565)	-2.8416	(0.0463)	-2.8579	(0.0170)	-2.8531
Quadratic $m(x) = \beta_0 + \beta_1^{ag} x_{ag} + \beta_1^{yr} x_{yr} + \beta_2^{ag} x_{ag}^2$							
	Fit to Subset III		Fit to Subset I		Fit to All Data		Observed
x_{ag}	$m_*^{(III)}$	$(s_*^{(III)})$	m_*^i	(s_*^i)	$m_*^{(All)}$	$(s_*^{(All)})$	μ
70	-3.7507	(0.0419)	-3.7711	(0.0332)	-3.7671	(0.0163)	-3.7630
80	-2.8774	(0.1046)	-2.8546	(0.0333)	-2.8553	(0.0164)	-2.8531

Table 6.5: GP model predictions for 2014 and Age 70/80 when fitted to various data subsets, cf. Table 6.2, indicated by superscripts. We report the predictive mean $m_*(x)$ and the predictive standard deviation $s_*^2(x_*) = C_*(x_*, x_*)$.

For another angle on forecasting with GP models, Figure 6.6 shows that the intercept-only model still performs well when predicting only slightly out-of-sample. In this Figure, we fitted mortality curves using the "notched" Subset II: years 1999–2010 and ages 50–84, plus 2011–2014 with ages 50–70, and then predicted out-of-sample for mortality rates for 2011–2014 and ages 71–85. This differs from the previous setup where the model had no prior information on

ages 71–84. We observed that in this setup the uncertainty from the intercept-only model is only slightly worse (wider interval) relative to the quadratic trend model, confirming the reasonableness of using the simpler $m(x) = \beta_0$.

Figure 6.6 also plots the marginal credible bands for $\mathbf{f}_*(\mathbf{x}_*)$ and intervals for future observations $\mathbf{y}_*$. As expected, the prediction uncertainty increases for the oldest ages and for later calendar years (compare credible intervals in Figure 6.6 for 2014 vis-a-vis 2011). Also note that the intervals for $\mathbf{y}_*$ are always a fixed distance away from the pointwise bands of $\mathbf{f}_*$ regardless of Age/Year due to the assumed constant noise variance σ^2; this is much more noticeable when in-sample, where posterior variance $C_*(\mathbf{x}_*, \mathbf{x}_*)$ is negligible relative to σ^2.

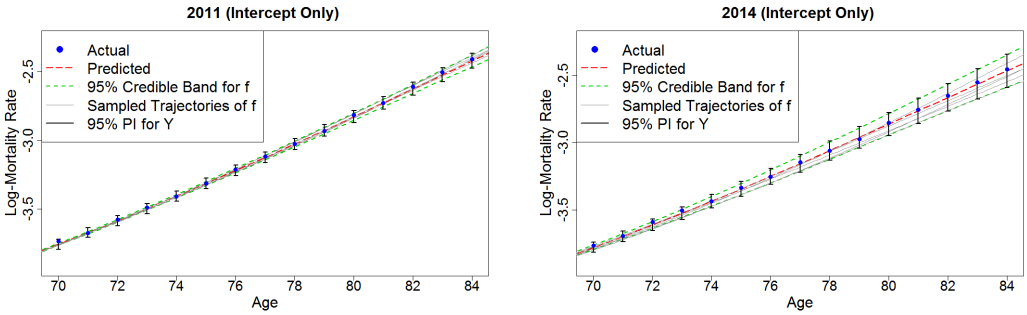

Figure 6.6: Mortality rate prediction for years 2011 and 2014 and ages 71–84. Model is fit with Subset II data with intercept-only mean functions and squared-exponential kernel. "Simulated paths of f" refers to simulated trajectories of the latent $\mathbf{f}_*$. Credible bands are for the mortality surface $\mathbf{f}_*$; vertical intervals are for predicted observable mortality experience $\mathbf{y}_*$.

As discussed, the GP model automatically generates credible intervals around any prediction, giving a principled approach for assessing uncertainty in forecasts. Moreover, since GP considers the full covariance structure of mortality curves, one can analytically evaluate the joint predictive uncertainty of any number of mortality rates. This is illustrated in Figure 6.6, where we generate a set of

trajectories (i.e. sample from the Gaussian multivariate predictive distribution) of log-mortality rates $\mathbf{f}_*(\mathbf{x}_*)$ for all ages $\mathbf{x}_*$ for a selected calendar year, namely 2014. Alternatively we could sample possible evolutions of mortality rates for a selected age, and a desired projection interval. Sampling such trajectories is crucial for quantifying aggregate mortality risk in a portfolio (say in a pension plan or life insurance context). Note that in contrast to factor models like Lee-Carter that force the mortality curve $\mu(\cdot, yr)$ to be confined to a low-dimensional space (e.g. one degree of freedom in classical Lee-Carter), within a GP framework, the shape of $f_*(\cdot, yr)$ remains non-parametric and infinite-dimensional.

For a further comparison, Figure 6.15 in Appendix B compares the predictions from a GP model against those of an age-period-cohort (APC) model (6.19). We observe that relative to the GP model, the APC model generates both volatile in-sample projections (as it is not designed with smoothing in mind), and erratic short-term projections due to the underlying time series fitted to the $\kappa_\cdot$ and $\gamma_\cdot$ factors. Recall that the APC framework tries to average out trends via a parametric model which makes the projections dependent even on distant historical experience, while the GP effectively uses the history to learn the spatial dependence structure and then makes data-driven projections based on recent experience. We note that both models perform poorly in long-range forecasting; this is to be expected with the GP model whose lengthscale θ_{yr} constrains reasonable extrapolation to 2-4 years out.

6.4.4 Forecasting Mortality Improvement

To focus more precisely on mortality *improvement*, we proceed to analyze changes in $\mu(x_{ag}, \cdot)$ over time. Section 6.4.1 discussed already backward-looking annual (YoY) improvements MI^{obs}_{back} and MI^{GP}_{back} as defined in Equation 6.10. For a more prospective analysis, one could consider a centered difference

$$1 - \left(\frac{\exp\left(f_*(x_{ag}, yr + h)\right)}{\exp\left(f_*(x_{ag}, yr - h)\right)} \right)^{1/2h} \approx - \frac{f_*(x_{ag}, yr + h) - f_*(x_{ag}, yr - h)}{2h}, \quad (6.12)$$

which is possible to compute for any h since the GP model for f_* yields an an entire mortality surface spanning over all $(x_{ag}, x_{yr}) \in \mathbb{R}^+ \times \mathbb{R}^+$. Note that since f_* is a Gaussian process, the right hand side of (6.12) remains Gaussian. We may also take the limit $h \to 0$ which gives the instantaneous rate of change of mortality in terms of calendar time. As an analogue to (6.12), we term the negative of the above differential as the instantaneous mortality improvement process

$$MI^{GP}_{diff}(x_{ag}; x_{yr}) \doteq - \frac{\partial f_*}{\partial x_{yr}}(x_{ag}, yr). \quad (6.13)$$

A remarkable property of the Gaussian process is that MI^{GP}_{diff} is once again a GP with explicitly computable mean and covariance functions (Williams and Rasmussen, 2006).

Proposition 1. *For the Gaussian Process f_* with a twice differentiable covariance kernel C, the limiting random variables*

$$\frac{\partial f_*}{\partial x_{yr}}(x_{ag}, yr) \doteq \lim_{h \to 0} \frac{f_*(x_{ag}, yr + h) - f_*(x_{ag}, yr)}{h} \quad (6.14)$$

exist in mean square and form a Gaussian process $\frac{\partial f_*}{\partial x_{yr}} \sim GP(m_{diff}, s_{diff})$. *Given the training set* $\mathcal{D} = (\mathbf{x}, \mathbf{y})$, *the posterior distribution of* $\frac{\partial f_*}{\partial x_{yr}}(x_*)$ *has mean and variance*

$$m_{diff}(x_*) = \mathbb{E}\left[\frac{\partial f_*}{\partial x_{yr}}(x_*)\middle| \mathbf{x}, \mathbf{y}\right] = \frac{\partial C}{\partial x'_{yr}}(\mathbf{x}, x_*)(\mathbf{C} + \mathbf{\Sigma})^{-1}\mathbf{y}, \tag{6.15}$$

$$s^2_{diff}(x_*) = \frac{\partial^2 C}{\partial x_{yr}\partial x'_{yr}}(x_*, x_*) - \frac{\partial C}{\partial x'_{yr}}(\mathbf{x}, x_*)(\mathbf{C} + \mathbf{\Sigma})^{-1}\frac{\partial C}{\partial x_{yr}}(x_*, \mathbf{x}), \tag{6.16}$$

where $\frac{\partial C}{\partial x'_{yr}}(\mathbf{x}, x_*) = \left[\frac{\partial C}{\partial x'_{yr}}(x^1, x_*), \ldots, \frac{\partial C}{\partial x'_{yr}}(x^N, x_*)\right]$ *and each component is computed as the partial derivative of* $C(x, x')$.

See Theorem 2.2.2 in Adler (2010) for more details. By analogy, Proposition 1 can also be extended to consider the differential of mortality to age or other covariates. Note that the squared exponential kernel in (6.8) is infinitely differentiable with derivatives

$$\frac{\partial C}{\partial x'_{yr}}(x, x') = -C(x, x')\frac{\eta^2}{\theta^2_{yr}}(x_{yr} - x'_{yr}), \tag{6.17}$$

$$\frac{\partial^2 C}{\partial x_{yr}\partial x'_{yr}}(x, x') = C(x, x')\frac{\eta^2}{\theta^2_{yr}}\left(1 - \frac{1}{\theta^2_{yr}}(x_{yr} - x'_{yr})^2\right). \tag{6.18}$$

Observe that the mean $m_{diff}(x_*)$ mortality improvement is equal to the derivative of the predicted mortality surface, $\frac{\partial}{\partial x_{yr}}m(x_*)$, a desirable self-consistency property. However, Proposition 1 goes much further, providing also analytic credible bands around $m_{diff}(x_*)$ and even the full predictive distribution of the mortality improvement process. Compare these features to a non-Bayesian smoothing model, such as P-splines, that only models $m(x_*)$ and therefore beyond direct differentiation provides no uncertainty quantification for $\frac{\partial f_*}{\partial x_{yr}}$.

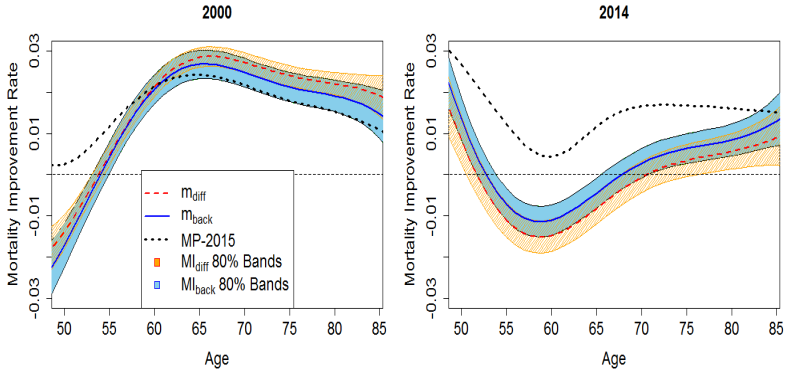

Figure 6.7: Estimated male mortality improvement using the differential GP model (instantaneous improvement) and the YoY improvement from the original GP model. We show the means and 80% credible bands for MI^{GP}_{diff} and MI^{GP}_{back} for males aged 50–84 and years 2000 & 2014. Models used are fit to All Data with $m(x) = \beta_0$.

To sum up the previous discussion, the GP framework yields a probabilistic estimate of the *instantaneous* mortality improvement which is analytically consistent with the projected mortality rates. Figure 6.7 shows mortality improvement estimates m^{GP}_{back}, m^{GP}_{diff} and MP-2015 improvement factors for ages 50–85 in years 2000 and 2014. The 80% credible bands of MI^{GP}_{back} and MI^{GP}_{diff} are also shown. The bands for MI^{GP}_{diff} were produced from (6.16), while for MI^{GP}_{back} they were generated from empirical sampling from (6.11). While we observe similar overall structure (in terms of similar predicted values and similar predicted uncertainty), we also note that there are some differences which indicate the changing rate of mortality improvement. Thus, in 2000, mortality improvement was accelerating, leading to $MI^{GP}_{diff}(\cdot; 2000) > MI^{GP}_{back}(\cdot; 2000)$. In contrast, the fact that $MI^{GP}_{diff}(\cdot; 2014) < MI^{GP}_{back}(\cdot; 2014)$ suggests that mortality improvement continues to decelerate as of 2014, so that the gap with the level improvement scale embedded in MP-2015 is likely to grow. In our analysis, we find that this deceleration

started around 2010, so that in the past 5-6 years mortality evolution over time has been convex, generating a growing wedge against the MP-2014/15 forecasts.

Remark. In our analysis we concentrate on modeling the log mortality surface, obtaining the mortality improvement factors as a by-product. An alternative is to first directly calculate observed mortality improvement MI_{back} and then model it with a GP. This would effectively replace the β_1^{yr} component of the mean function with a richer structure. This procedure is similar to that of Mitchell et al. (2013) where mortality improvement itself is modeled in a Lee-Carter framework.

6.5 Extensions of GP Models

6.5.1 Inhomogeneous GP Models

Basic GP assumes a stationary covariance structure which may not be appropriate. If the spatial dependence in mortality experience is state-dependent, i.e. $C(x^i, x^j)$ depends on x^i, x^j (and not just $|x^i - x^j|$), this would introduce model misspecification and lead to poor model performance (i.e. too much or too little smoothing).

To test for inhomogeneous correlation, we consider a GP model segmented by age. This means that we introduce a piecewise setup, fitting three different GP models depending on x_{ag}. The age grouping was done manually according to (younger) $x_{ag} \in \{50, \ldots, 69\}$, (older) $x_{ag} \in \{70, \ldots, 84\}$, as well as the full model $x_{ag} \in \{50, \ldots, 84\}$, and an extended model considering all ages $x_{ag} \in \{1, \ldots, 84\}$. Table 6.6 presents the fitted trend and hyper-parameters for each group using a model fitted to all years 1999–2014 and quadratic mean function.

Ages Fit	β_0	β_1^{ag}	β_2^{ag}	β_1^{yr}
Extended $[1, 84]$	-23.533	-0.005	8.402e-04	7.797e-03
Younger $[50, 69]$	10.521	0.084	-3.336e-05	-9.908e-03
Older $[70, 84]$	26.806	-0.016	7.113e-04	-1.635e-02
All $[50, 84]$	19.336	0.041	3.324e-04	-1.367e-02
Ages Fit	η^2	σ^2	θ_{ag}	θ_{yr}
Extended $[1, 84]$	1.904e-01	1.184e-03	3.966	12.795
Younger $[50, 69]$	2.633e-03	2.964e-04	4.501	4.196
Older $[70, 84]$	1.489e-03	1.517e-04	14.709	6.661
All $[50, 84]$	1.760e-03	2.336e-04	4.543	3.825

Table 6.6: GP models fitted by age groups. All models are fitted to years 1999–2014 and using a squared-exponential kernel with a quadratic mean function $m(x^n) = \beta_0 + \beta_1^{ag} x_{ag}^n + \beta_1^{yr} x_{yr}^n + \beta_2^{ag} (x_{ag}^n)^2$. The reported hyper-parameter values are maximum likelihood estimates from `DiceKriging`.

Table 6.6 shows that the Extended age group trend/shape parameter estimates differ from the remaining groups, likely due to the fact that infant and adolescent mortality produce a non-quadratic mortality shape in age. Furthermore, the respective positive coefficient of the Extended β_1^{yr} parameter contradicts the idea of mortality improvement and possibly indicates poor goodness-of-fit.

Segmenting the older ages does generate some reasonable differences in fitted models: log-mortality is linear in the younger group, so that the β_2^{ag} coefficient is negligible; it is larger in the older age group due to the rapid increase of mortality in age; combining the two as was done originally yields an average of the two estimates. The estimates of β_{yr}^1 also support the claim of Older mortality improving faster than Younger mortality: log-mortality decreases annually at 1% for for the Younger group and at 1.6% for the Older group. The θ_{yr} values are all similar across groups, except for the Extended group which needs to compensate for its poor trend fit. The Younger and Extended fits share similar θ_{ag} values. We

attribute the larger θ_{ag} for Older ages to fitting issues due to a complicated age dependence and only 15 ages worth of data (it could also suggest that mortality rates of older ages are more correlated). A similar effect happens for female data where $\theta_{ag}^{Fem} = 44.118$ for Older ages, see Table 6.7 in Section 6.7.

In sum, a "global" model which includes all ages is inappropriate due to the much younger ages having vastly different mean and covariance structures. An improved fit is potentially possible through segmenting the Ages into subgroups, but we encounter issues due to the datasets becoming too small, hurting credibility. Recall that the precise age groups were picked manually and a more detailed "change-point" analysis may be warranted to determine the best segmentation of data, and whether the lower cutoff at age 50 is appropriate. We remark that there exist hierarchical GP models (Gramacy and Taddy, 2012) that attempt to automatically carry out such data splitting.

6.5.2 Modeling Cause of Death Scales

The raw CDC data are classified by cause of death and hence it is in fact possible to build a comprehensive mortality improvement model that is broken down beyond the basic Male/Female distinction. Understanding the different trends in cause-of-death can be important as there has been uneven progress (and in some situations reversal) of longevity improvements by cause. For example, the large improvement in mortality from coronary artery disease has not been matched by improvements in mortality from cancer. Different causes of death affect different ages, creating multiple "cross-currents" that drive mortality, a fact which is important for long-term projections.

Thus, mortality improvement models can benefit from analyzing by-cause data. Building such models would need to balance the risk of over-specification with the benefit of incorporating additional data. Key issues and concepts in building a by-cause model are:

- The mean function, m, would need to be fit to each cause.
- The covariance function controlling spatial correlation would also likely differ by cause.
- This paper focuses on modeling the log mortality rate. A by-cause model would benefit instead from modeling the force of mortality from each cause, as the total force of mortality is simply a sum of the underlying by-causes forces of mortality. However this additive structure does not match the log transformation applied in this paper.
- Bayesian models with informative priors for mean function and other coefficients would provide a degree of protection against overfitting the models.
- A hierarchical model which builds in a relationship between the by-cause trend coefficients could be tested.

Such analysis is left for further research.

6.5.3 Model Updating

The GP model is convenient for analysis when new data becomes available. This is in contrast to methods, such as splines, which require a full model refit. With GPs, once the correlation structure is fit (and assuming it did not change), the Gaussian posterior $\mathbf{f}_*$ allows for an updated $\mathbf{m}_*$ and $\mathbf{C}_*$, see Ludkovski (2015,

Section 5.1) for details. These formulas showcase the explicit impact of additional data, both for smoothing past experience, or projecting forward in time.

To illustrate the effect of a new year of data, we compute the predicted mean $\mathbf{m}_*$ and standard deviation $\mathbf{C}_*$ for age 65 and years 1999, 2013 and 2016, first based on data for all ages and calendar years 1999–2013, and then updated with year-2014 data. The results are listed in Table 6.5.3.

	Before Updating (1999–2013)		After Updating (1999–2014)	
x_{yr}	$\mathbb{E}[f(65, x_{yr})\|\mathbf{x}, \mathbf{y}]$	$s_*(65, x_{yr})$	$\mathbb{E}[f(65, x_{yr})\|\tilde{\mathbf{x}}, \tilde{\mathbf{y}}]$	$\tilde{s}_*(65, x_{yr})$
1999	-3.8845	0.0174	-3.8849	0.0173
2013	-4.1497	0.0174	-4.1502	0.0170
2016	-4.1197	0.0266	-4.1248	0.0208

Table 6.7: GP model updating: $\mathbf{x}, \mathbf{y}$ refers to observed mortality for ages 50–84, years 1999–2013; $\tilde{\mathbf{x}}, \tilde{\mathbf{y}}$ is the same data augmented with year-2014 experience. The mean function is intercept-only, $m(x) = \beta_0$; s_* is posterior standard deviation.

The additional year of credibility decreases posterior standard deviations s_*. Unsurprisingly, the impact on 1999-prediction is negligible since it is so far in the past. The standard deviation for 2013 has a slight decrease after updating, while 2016 has a much larger reduction: the original model was initially predicting 3 years out-of-sample, while the updated one does for just 2 years out-of-sample. Similarly, the in-sample means change only slightly, while the out-of-sample 2016 has a larger adjustment. The overall decrease in updated posterior means is consistent with the fact that the observed log-mortality for age 65 in 2014 was -4.1543, lower than the predicted -4.1443 using the 1999–2013 model.

6.5.4 Other Extensions

A standard assumption is that mortality curves are increasing in Age, i.e. $x_{ag} \mapsto f(x_{ag}, \cdot)$ is monotone. The basic GP framework does not impose any monotonicity restriction. Such structural constraints on f can help in improving mortality projection in terms of m_* (especially for long-range forecasts), as well as reduce predictive uncertainty measured by s_*^2. At the same time, constraints are at odds with the underlying Gaussian random field statistical paradigm, introducing additional complexity in fitting and making inference from the constrained posterior.

One promising recent solution was proposed in Riihimäki and Vehtari (2010) who suggested incorporating monotonicity by adding virtual observation points $\tilde{x}_i, \tilde{m}_i$ for the derivative of $f(x_i)$. Because the derivative $\boldsymbol{f}'$ also forms a GP, one can explicitly write down the joint covariance structure of $(\boldsymbol{f}, \boldsymbol{f}')$ (for example the posterior mean of $\boldsymbol{f}'$ is the derivative of m_*). Monotonicity is then implied by requiring the derivative to be positive at the given $\tilde{x}_i$'s. As the size of the latter collection increases, the resulting estimate is more and more likely to be increasing *everywhere* in the domain. This strategy circumvents the direct monotonicity restriction while maintaining computational tractability through linear constraints. Riihimäki and Vehtari (2010) give a recipe for adaptively placing such virtual derivative points by iteratively adding new $\tilde{x}_i$'s where the current m_* violates monotonicity. Further constraints, such as expert opinions about mortality at extreme ages (100+) could be beneficially added.

An additional extension involves use of multiple data sets; there are many instances where mortality data from one source might be more up-to-date than from other sources, for example CDC data provides at least 3 more years of

information than SSA data. The use of co-kriging models or the use of CDC data as an input to a GP used to model SSA data is another avenue of possible future research. Such co-kriging models might also be helpful when using population improvement data to supplement a GP analysis of a specific insurance company's or pension fund's mortality experience.

6.6 Conclusion

We have proposed and investigated the use of Gaussian Process models for smoothing and forecasting mortality surfaces. Our approach takes a unified view of the mortality experience as a statistical *response surface* that is noisily reflected in realized mortality experience. A statistical procedure is then used to calibrate the spatial dependence among the latent log-mortality rates. The GP model provides a consistent, non-parametric framework for uncertainty quantification in *both* the mortality surface itself, as well as mortality improvement, which corresponds to relationship between f and x_{yr}. This quantification can be done in-sample, by retrospectively smoothing raw mortality counts, or out-of-sample, by building mean forecasts, uncertainty bands, and full scenarios for future mortality/mortality improvement evolution. In contrast, traditional actuarial techniques for graduating data (e.g. the Whittaker-Henderson model used by RPEC) focus on smoothing noisy data but fail to provide measures of uncertainty about the fit.

We have focused on population data and smoothing over age and year. The model can be easily extended to additional dimensions, such as duration and net worth in the context of life insurance, or year-of-birth cohort for pension mortality

analysis. Adding covariates to the definition of the covariance kernel $C(x, x')$ is straightforward, with the main challenge lying in interpreting the resulting GP parameters which would reflect a modified concept of spatial distance.

Perhaps the most useful application of our model is for analyzing the latest mortality data, i.e. at the "edge" of the mortality surface. Here we find and document the statistical evidence that US mortality improvements have materially moderated across a large swath of ages. In particular, for Ages 55–70, US mortality has been effectively flat, or possibly even increasing in the 2010's. This points to a large divergence from the MP-2015 improvement scales that continue to assume significant mortality gains for all ages and would seem to be overstated at least in the near-term. Moreover, by explicitly computing the differential mortality improvement MI^{GP}_{diff}, our model gives the most current, instantaneous forecast on mortality improvement, in contrast to the traditional year-over-year estimates.

On a related note, our analysis quantifies the apparent correlation in observed mortality experience across Age and calendar Year. Thus, the obtained estimates of length-scales θ_{yr}, imply that studies with very long historical analysis (e.g. going back to 1950 or even 1900) may not add much value to our understanding of current or future projected trends in mortality improvement. Similarly, long-term projections of future mortality improvement (e.g. MP-2015 which is used for projecting mortality often 40 to 60 years into the future) contain a higher degree of uncertainty than is typically recognized in actuarial analyses. Indeed, our results suggest that projections more than a decade into the future are entirely based on the assumed prior calendar trend and hence have almost no credibility based on observed experience.

Our results show that even a "vanilla" implementation of a GP model already produces useful statistical description of the mortality experiences that is competitive with existing methods in terms of its probabilistic richness and accuracy. We therefore see an enormous potential for further works in this direction, in particular to resolve some further methodological challenges. Mean function modeling which is typically not an important component of GP models in other contexts, is critical for actuaries when projecting out-of-sample. Also, constrained GP models that structurally enforce the age-shape of mortality could be promising in creating better future forecasts. Yet another challenge is better blending of the data-influenced prediction and the prior mean for extrapolation which can be achieved with other Gaussian field specifications or other techniques (Salemi et al., 2013; Lee and Owen, 2015). A different challenge consists in creating meaningful backtesting analyses which would test not just predictive accuracy of m_*, but also the quality of the generated credibility intervals (both for mortality rates and mortality improvements), and the assumption of Age- and Year-stationary covariance structure. On that point, it would be worthwhile to investigate data from other countries to infer commonalities in mortality correlations.

6.7 Appendix: Tables and Figures for Female Data

In this section we list the female counterpart of the tables and figures above associated to male data. In general, female mortality is lower, but the Age-shapes are mostly the same. For smoothing, Figures 6.1 and 6.8 are nearly identical in

shape. The curve in Figure 6.9 for 2014 is slightly different in shape compared to the male Figure 6.3 around ages 50–65 due to the observed mortality improvement declining in this range. Comparing annual mortality improvement in Figures 6.4 and 6.10, the female data shows slightly lower improvement overall.

Comparing Tables 6.8 with 6.4 and Figures 6.11 with 6.5, we see that the trend model comparison results are near identical; the only noticeable differences are that the quadratic model is a much better fit on the test set for females, and that the θ values for the intercept-only model are larger.

The intercept-only parameters for both trend function and GP for females in Table 6.7 are nearly identical to those for males in Table 6.6 as well as the GP parameters for the quadratic model. The quadratic trend function parameters are much different for males and females. In particular, the intercept terms are all much different in magnitude, and some of the higher order terms differ in sign. This is likely an indication that the trend curves differs in shape between genders in their respective age groups, which is unsurprising since the age group endpoints were chosen to match the male dataset.

Figure 6.2 already showed both male and female mortality improvement over time for ages 60, 70 and 84, which explains the shape differences in Figures 6.13 and 6.7. As with the male data, we still observe $MI^{GP}_{diff} > MI^{GP}_{back}$ in 2000 and the reverse in 2014 which was explained due to mortality acceleration in Section 6.4.4.

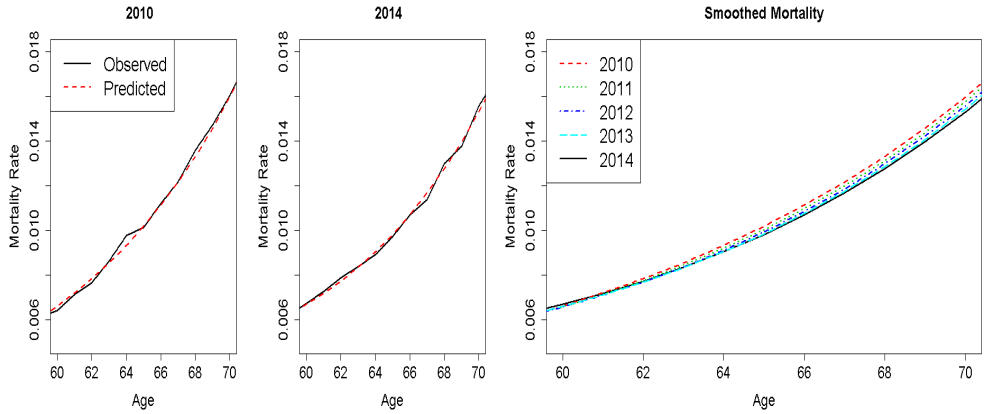

Figure 6.8: Mortality rates for Females aged 60–70 during years 2010–2014. Raw vs. estimated smoothed mortality curves. Models are fit to All Female data.

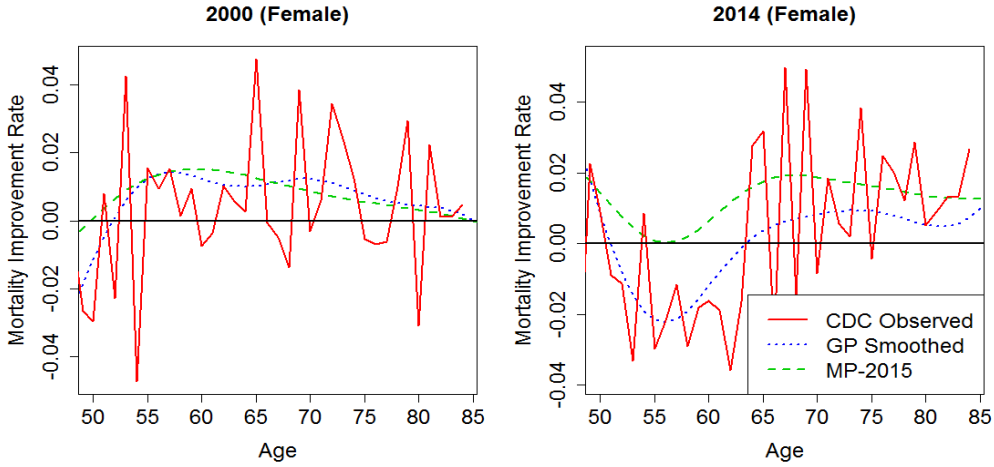

Figure 6.9: Mortality improvement factors for Females using All data. Solid red lines indicate the empirical mortality experience; dotted blue lines are the smoothed estimates using a GP, dashed green lines are the published MP-2015 improvement factors.

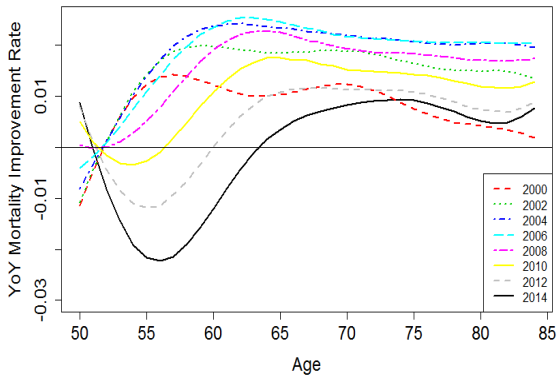

Figure 6.10: Comparison of yearly mortality improvement factors for Females using All data. The curve for 2014 is the same as in Figure 6.9.

	Mean Function Parameter MLE's			
	β_0	β_1^{ag}	β_2^{ag}	β_1^{yr}
Intercept	-5.101	-	-	-
Linear	4.484	0.083	-	-7.167e-03
Quadratic	11.207	0.054	2.712e-04	-1.014e-02
	GP Hyperparameter MLE's			
	η^2	σ^2	θ_{ag}	θ_{yr}
Intercept	4.444e-01	2.968e-04	7.363	10.882
Linear	2.802e-03	3.682e-04	4.432	4.505
Quadratic	2.053e-03	2.911e-04	4.464	4.384

Table 6.8: Mean functions and fitted covariance parameters using Set I Female Data (ages 50–70 and years 1999–2010). The mean functions are $m(x) = \beta_0$ for Intercept, $m(x) = \beta_0 + \beta_1^{ag} x_{ag} + \beta_1^{yr} x_{yr}$ for Linear, and $m(x) = \beta_0 + \beta_1^{ag} x_{ag} + \beta_1^{yr} x_{yr} + \beta_2^{ag} x_{ag}^2$ for Quadratic.

6.8 Appendix: Supplementary Plots

6.8.1 GP Model Residuals

Figure 6.14 provides a Q-Q plot of the GP model (fitted to the full dataset) residuals. For contrast, the Lee-Carter model as in (6.2) is included. Under the latter framework, modelers typically assume a non-Gaussian noise structure. The

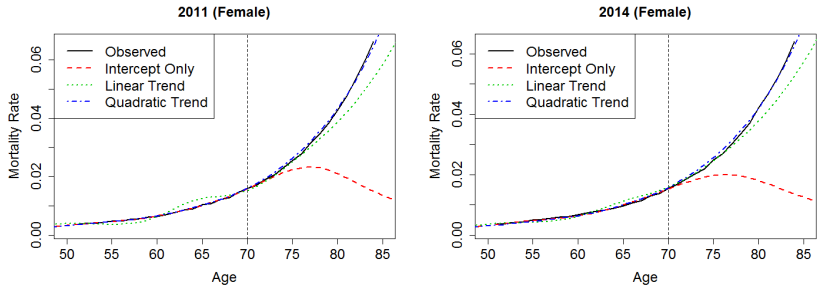

Figure 6.11: Comparison of mean function choices in extrapolating mortality rates at old ages for Females. Models are fit to years 1999–2010 and ages 50–70 (Subset III), with estimates made for Age 50–85 in 2011 and 2014. The vertical line indicates the boundary of the training dataset in x_{ag}. The mean functions are given in Table 6.8.

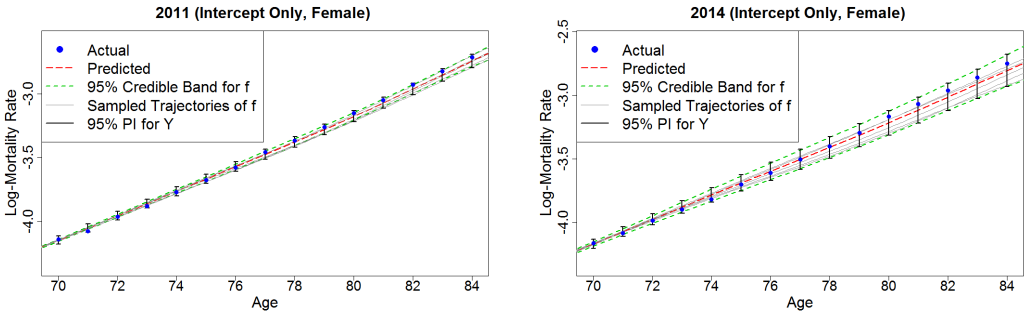

Figure 6.12: Mortality rate prediction for years 2011 and 2014 and ages 71–84. Model is fit on Subset II Female data with intercept-only mean function and squared-exponential kernel.

figure suggests that the GP residuals are reasonably Gaussian, with mildly heavy tails. It also agrees with the lack of normality of a Lee-Carter fit, where the plot indicates both skew and heavier tails. We remark that the GP framework can be extended to cover other observation models, such as binomial which could be closer in line with (6.7). However this comes with significant computational costs, as well as reducing interpretability, which we deem to not be worth a minimal improvement in assumptions realism.

Ages Fit	β_0	β_1^{ag}	β_2^{ag}	β_1^{yr}
Extended $[1, 84]$	-25.224	-0.008	8.721e-04	8.678e-03
Younger $[50, 69]$	1.128	0.080	3.912e-05	-5.471e-03
Older $[70, 84]$	17.272	-0.038	9.071e-04	-1.151e-02
All $[50, 84]$	7.473	0.035	4.186e-04	-7.980e-03
Ages Fit	η^2	σ^2	θ_{ag}	θ_{yr}
Extended $[1, 84]$	2.170e-01	1.187e-03	4.095	13.040
Younger $[50, 69]$	4.311e-03	2.907e-04	5.695	5.487
Older $[70, 84]$	2.543e-03	1.334e-04	44.118	6.856
All $[50, 84]$	2.814e-03	2.236e-04	5.574	5.249

Table 6.9: GP models fitted by age groups with Female data. All models used squared-exponential kernel and years 1999–2014.

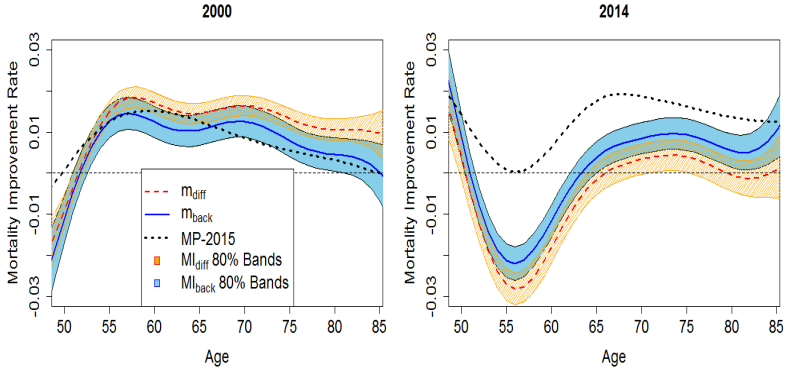

Figure 6.13: Estimated Female mortality improvement using the differential GP model (instantaneous improvement) and the YoY improvement from the original GP model. We show the means and 80% uncertainty bands for MI^{GP}_{diff} and MI^{GP}_{back} for Females aged 50–84 and years 2000 & 2014. Models used are fit to All Data .

Comparison of GP and APC forecasts

To provide a brief comparison of the popular stochastic mortality models, we fit a cohort extension of the Lee-Carter model in (6.2), introduced by Renshaw

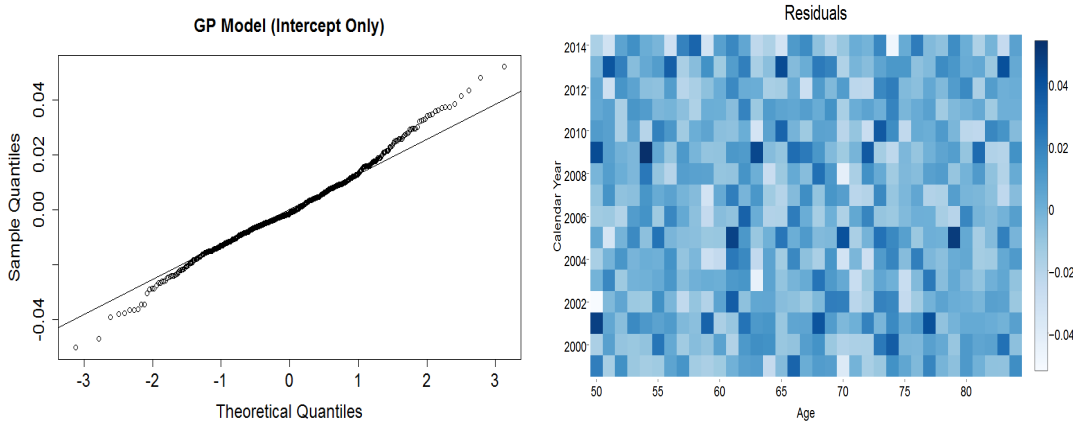

Figure 6.14: Left: Q-Q Plots for residuals of a fitted GP model with mean function $m(x) = \beta_0$. We use Male All Data to test the normality assumption of ϵ in (6.3). We observe that the GP residuals are reasonably Gaussian with mildly heavy tails. Right: heatmap of $\epsilon(x)$ as a function of the two-dimensional input $x = (x_{ag}, x_{yr})$. We observe no apparent correlation in the fitted residuals.

and Haberman (2006), which is as follows:

$$\mu_{ij} = \alpha_i + \frac{1}{n_a}\kappa_j + \frac{1}{n_a}\gamma_{j-i} + \epsilon_{ij}, \tag{6.19}$$

where γ_{j-i} is the cohort effect and n_a is the number of years in the data set. Using the `StMoMo` software suite (Villegas et al., 2015b) on our data yielded a random walk with drift for κ_j and ARIMA(0,1,2) model for γ_{j-i}. Cairns et al. (2011a) showed that this model performed well in US male data analysis. The results are shown in Figure 6.15 where we predict for two representative ages and for years 1999–2020 (i.e. both in-sample and up to 5 years into the future).

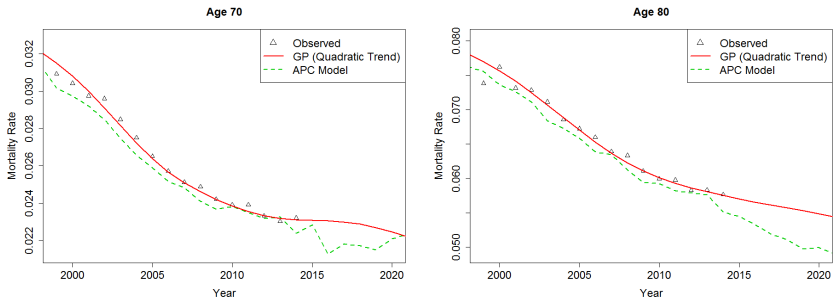

Figure 6.15: Observed and predicted mortality rates. GP model uses quadratic mean function $m(x) = \beta_0 + \beta_1^{ag} x_{ag} + \beta_1^{yr} x_{yr} + \beta_2^{ag} x_{ag}^2$, and the APC model is as in Equation (6.19).

Bibliography

Adler, R. J., 2010. The geometry of random fields. Vol. 62 of Classics in Applied Mathematics. SIAM.

Ankenman, B., Nelson, B. L., Staum, J., 2010. Stochastic kriging for simulation metamodeling. Operations research 58 (2), 371–382.

Bacinello, A., Biffis, E., Millossovich, P., 2010. Regression-based algorithms for life insurance contracts with surrender guarantees. Quantitative Finance 10 (9), 1077–1090.

Bacinello, A. R., Millossovich, P., Olivieri, A., Pitacco, E., 2011. Variable annuities: A unifying valuation approach. Insurance: Mathematics and Economics 49 (3), 285–297.

Ballotta, L., Haberman, S., 2006. The fair valuation problem of guaranteed annuity options: The stochastic mortality environment case. Insurance: Mathematics and Economics 38 (1), 195–214.

Barrieu, P., Bensusan, H., El Karoui, N., Hillairet, C., Loisel, S., Ravanelli, C., Salhi, Y., 2012. Understanding, modelling and managing longevity risk: key issues and main challenges. Scandinavian Actuarial Journal 2012 (3), 203–231.

Bauer, D., Benth, F. E., Kiesel, R., 2012a. Modeling the forward surface of mortality. SIAM Journal on Financial Mathematics 3 (1), 639–666.

Bauer, D., Kling, A., Russ, J., 2008. A universal pricing framework for guaranteed minimum benefits in variable annuities. Astin Bulletin 38 (02), 621–651.

Bauer, D., Reuss, A., Singer, D., 2012b. On the calculation of the solvency capital requirement based on nested simulations. ASTIN Bulletin 42 (02), 453–499.

Bect, J., Ginsbourger, D., Li, L., Picheny, V., Vazquez, E., 2012. Sequential design of computer experiments for the estimation of a probability of failure. Statistics and Computing 22 (3), 773–793.

Binois, M., Gramacy, R. B., Ludkovski, M., 2016. Practical heteroskedastic gaussian process modeling for large simulation experiments. arXiv preprint arXiv:1611.05902.

Booth, H., Maindonald, J., Smith, L., 2002. Applying Lee–Carter under conditions of variable mortality decline. Population Studies 56 (3), 325–336.

Bretthauer, K. M., Ross, A., Shetty, B., 1999. Nonlinear integer programming for optimal allocation in stratified sampling. European Journal of Operational Research 116 (3), 667–680.

Broadie, M., Du, Y., Moallemi, C. C., 2011. Efficient risk estimation via nested sequential simulation. Management Science 57 (6), 1172–1194.

Brooks, S., Gelman, A., Jones, G., Meng, X.-L., 2011. Handbook of Markov Chain Monte Carlo. CRC press.

Brouhns, N., Denuit, M., Vermunt, J. K., 2002a. A Poisson log-bilinear regression approach to the construction of projected lifetables. Insurance: Mathematics and Economics 31 (3), 373–393.

Brouhns, N., Denuit, M., Vermunt, J. K., 2002b. A Poisson log-bilinear regression approach to the construction of projected lifetables. Insurance: Mathematics and Economics 31 (3), 373–393.

Cairns, A. J., Blake, D., Dowd, K., 2006. A two-factor model for stochastic mortality with parameter uncertainty: Theory and calibration. Journal of Risk and Insurance 73 (4), 687–718.

Cairns, A. J., Blake, D., Dowd, K., Coughlan, G. D., Epstein, D., Khalaf-Allah, M., 2011a. Mortality density forecasts: An analysis of six stochastic mortality models. Insurance: Mathematics and Economics 48 (3), 355–367.

Cairns, A. J., Blake, D., Dowd, K., Coughlan, G. D., Epstein, D., Ong, A., Balevich, I., 2009a. A quantitative comparison of stochastic mortality models using data from England and Wales and the United States. North American Actuarial Journal 13 (1), 1–35.

Cairns, A. J., Blake, D., Dowd, K., Coughlan, G. D., Epstein, D., Ong, A., Balevich, I., 2009b. A quantitative comparison of stochastic mortality models using data from England and Wales and the United States. North American Actuarial Journal 13 (1), 1–35.

Cairns, A. J., Blake, D., Dowd, K., Coughlan, G. D., Khalaf-Allah, M., 2011b. Bayesian stochastic mortality modelling for two populations. ASTIN Bulletin 41 (01), 29–59.

Cairns, A. J., Dowd, K., Blake, D., Coughlan, G. D., 2014. Longevity hedge effectiveness: A decomposition. Quantitative Finance 14 (2), 217–235.

Cairns, A. J., El Boukfaoui, G., 2017. Basis risk in index based longevity hedges: A guide for longevity hedgers. Tech. rep., Working paper. Edinburgh: Heriot-Watt University.

Camarda, C. G., 2012. Mortalitysmooth: An R package for smoothing Poisson counts with P-splines. Journal of Statistical Software 50 (1), 1–24.

Carpenter, B., Lee, D., Brubaker, M. A., Riddell, A., Gelman, A., Goodrich, B., Guo, J., Hoffman, M., Betancourt, M., Li, P., 2016. Stan: A probabilistic programming language. Journal of Statistical Software to Appear.

Chan, T. F., Golub, G. H., LeVeque, R. J., 1982. Updating formulae and a pairwise algorithm for computing sample variances. In: COMPSTAT 1982 5th Symposium held at Toulouse 1982. Springer, pp. 30–41.

Chauvigny, M., Devineau, L., Loisel, S., Maume-Deschamps, V., 2011. Fast remote but not extreme quantiles with multiple factors: applications to solvency ii and enterprise risk management. European Actuarial Journal 1 (1), 131–157.

Chen, H., Cox, S. H., 2009. Modeling mortality with jumps: Applications to mortality securitization. Journal of Risk and Insurance 76 (3), 727–751.

Chen, L., 1996. Stochastic mean and stochastic volatility: a three-factor model of the term structure of interest rates and its applications in derivatives pricing and risk management. Blackwell publishers.

Chen, X., Nelson, B. L., Kim, K.-K., 2012. Stochastic kriging for conditional value-at-risk and its sensitivities. In: Proceedings of the 2012 Winter Simulation Conference (WSC). IEEE, pp. 1–12.

Chevalier, C., Bect, J., Ginsbourger, D., Vazquez, E., Picheny, V., Richet, Y., 2014a. Fast parallel kriging-based stepwise uncertainty reduction with application to the identification of an excursion set. Technometrics 56 (4), 455–465.

Chevalier, C., Picheny, V., Ginsbourger, D., 2014b. Kriginv: An efficient and user-friendly implementation of batch-sequential inversion strategies based on kriging. Computational Statistics & Data Analysis 71, 1021–1034.

Christiansen, M. C., Niemeyer, A., 2014. Fundamental definition of the solvency capital requirement in solvency ii. ASTIN Bulletin: The Journal of the IAA 44 (3), 501–533.

Committee, B., et al., 2013. Fundamental review of the trading book: A revised market risk framework. Consultative Document, October.

Continuous Mortality Investigation, 2015. The CMI mortality projections model, CMI 2015. Tech. rep., CMI Working Paper 84.
URL http://www.actuaries.org.uk/learn-and-develop/continuous-mortality-investigation/cmi-working-papers/mortality-projections/cmi-wp-84

Coughlan, G. D., Khalaf-Allah, M., Ye, Y., Kumar, S., Cairns, A. J., Blake, D., Dowd, K., 2011. Longevity hedging 101: A framework for longevity basis risk analysis and hedge effectiveness. North American Actuarial Journal 15 (2), 150–176.

Currie, I. D., 2013. Smoothing constrained generalized linear models with an application to the Lee-Carter model. Statistical Modelling 13 (1), 69–93.

Currie, I. D., 2016. On fitting generalized linear and non-linear models of mortality. Scandinavian Actuarial Journal 2016 (4), 356–383.

Currie, I. D., Durban, M., Eilers, P. H., 2004. Smoothing and forecasting mortality rates. Statistical Modelling 4 (4), 279–298.

Czado, C., Delwarde, A., Denuit, M., 2005a. Bayesian Poisson log-bilinear mortality projections. Insurance: Mathematics and Economics 36 (3), 260–284.

Czado, C., Delwarde, A., Denuit, M., 2005b. Bayesian poisson log-bilinear mortality projections. Insurance: Mathematics and Economics 36 (3), 260–284.

Debón, A., Martínez-Ruiz, F., Montes, F., 2010. A geostatistical approach for dynamic life tables: The effect of mortality on remaining lifetime and annuities. Insurance: Mathematics and Economics 47 (3), 327–336.

Debonneuil, E., 2010. A simple model of mortality trends aiming at universality: Lee Carter + Cohort. Tech. rep., arXiv:1003.1802.

Delwarde, A., Denuit, M., Eilers, P., 2007. Smoothing the Lee–Carter and Poisson log-bilinear models for mortality forecasting a penalized log-likelihood approach. Statistical Modelling 7 (1), 29–48.

Dhaene, J., Vanduffel, S., Goovaerts, M., Kaas, R., Tang, Q., Vyncke, D., 2006. Risk measures and comonotonicity: a review. Stochastic models 22 (4), 573–606.

Dokumentov, A., Hyndman, R. J., 2014. Bivariate data with ridges: two-dimensional smoothing of mortality rates. Tech. rep., Working paper series, Monash University.

Fang, K.-T., Li, R., Sudjianto, A., 2005. Design and modeling for computer experiments. CRC Press.

Forrester, A., Sobester, A., Keane, A., 2008. Engineering design via surrogate modelling: a practical guide. John Wiley & Sons.

Fushimi, T., Kogure, A., 2014. A Bayesian approach to longevity derivative pricing under stochastic interest rates with a two-factor Lee-Carter model. Tech. rep., ARIA 2014 Annual Meeting.
URL http://www.aria.org/Annual_Meeting/2014/2014_Accepted_Papers/4C/FushimiandKogure.pdf

Gan, G., Lin, X. S., 2015. Valuation of large variable annuity portfolios under nested simulation: A functional data approach. Insurance: Mathematics and Economics 62, 138–150.

Girosi, F., King, G., 2008. Demographic forecasting. Princeton University Press.

Gobet, E., Lemor, J.-P., Warin, X., et al., 2005. A regression-based monte carlo method to solve backward stochastic differential equations. The Annals of Applied Probability 15 (3), 2172–2202.

Golub, G. H., Van Loan, C. F., 2012. Matrix computations. Vol. 3. JHU Press.

Gordy, M. B., Juneja, S., 2010. Nested simulation in portfolio risk measurement. Management Science 56 (10), 1833–1848.

Gramacy, R., Taddy, M., 2012. Tgp, an R package for treed Gaussian process models. Journal of Statistical Software 33, 1–48.

Gramacy, R. B., Lee, H. K., 2008. Gaussian processes and limiting linear models. Computational Statistics & Data Analysis 53 (1), 123–136.

Gramacy, R. B., Ludkovski, M., 2015. Sequential design for optimal stopping problems. SIAM Journal on Financial Mathematics 6 (1), 748–775.

Harrell, F. E., Davis, C., 1982. A new distribution-free quantile estimator. Biometrika 69 (3), 635–640.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The elements of statistical learning, 2nd Edition. Springer Series in Statistics. Springer.

Helbert, C., Dupuy, D., Carraro, L., 2009. Assessment of uncertainty in computer experiments from Universal to Bayesian Kriging. Applied Stochastic Models in Business and Industry 25 (2), 99–113.

Hunt, A., Blake, D., 2014. A general procedure for constructing mortality models. North American Actuarial Journal 18 (1), 116–138.

Hyndman, R. J., 2015. forecast: Forecasting functions for time series and linear models. R package version 6.1.
URL `http://github.com/robjhyndman/forecast`

Hyndman, R. J., Ullah, M. S., 2007a. Robust forecasting of mortality and fertility rates: A functional data approach. Computational Statistics & Data Analysis 51 (10), 4942–4956.

Hyndman, R. J., Ullah, M. S., 2007b. Robust forecasting of mortality and fertility rates: a functional data approach. Computational Statistics & Data Analysis 51 (10), 4942–4956.

Jalen, L., Mamon, R., 2009. Valuation of contingent claims with mortality and interest rate risks. Mathematical and Computer Modelling 49 (9), 1893–1904.

Johnson, L. R., Gramacy, R. B., Cohen, J., Mordecai, E., Murdock, C., Rohr, J., Ryan, S. J., Stewart-Ibarra, A. M., Weikel, D., 2017. Phenomenological forecasting of disease incidence using heteroskedastic gaussian processes: a dengue case study. arXiv preprint arXiv:1702.00261.

Kamiński, B., 2015. A method for the updating of stochastic kriging metamodels. European Journal of Operational Research 247 (3), 859–866.

Kessler, A., McCloskey, W., Bensoussan, A., 2015. The pension risk transfer market at $240 billion: Innovation, globalization, and growth. Special Issues 2015 (1), 18–27.

Kim, J. H. T., Hardy, M. R., 2007. Quantifying and correcting the bias in estimated risk measures. Astin Bulletin 37 (02), 365–386.

Kimeldorf, G., Wahba, G., 1971. Some results on tchebycheffian spline functions. Journal of mathematical analysis and applications 33 (1), 82–95.

Kleijnen, J. P., 2007. Design and analysis of simulation experiments. Vol. 111. Springer Science & Business Media.

Krige, D. G., 1951. A statistical approach to some basic mine valuation problems on the witwatersrand. Journal of the Southern African Institute of Mining and Metallurgy 52 (6), 119–139.

Labopin-Richard, T., Picheny, V., 2016. Sequential design of experiments for estimating percentiles of black-box functions. arXiv preprint arXiv:1605.05524.

Lee, M. R., Owen, A. B., 2015. Single nugget kriging. Tech. rep., arXiv preprint arXiv:1507.05128.

Lee, R., Miller, T., 2001. Evaluating the performance of the Lee–Carter method for forecasting mortality. Demography 38 (4), 537–549.

Lee, R. D., Carter, L. R., 1992. Modeling and forecasting US mortality. Journal of the American Statistical Association 87 (419), 659–671.

Li, H., O'Hare, C., 2015. Mortality forecast: Local or global? Tech. rep., Available at SSRN 2612420.

Li, J.-H., Hardy, M., Tan, K., 2009. Uncertainty in model forecasting: An extension to the classic Lee-Carter approach. ASTIN Bulletin 39, 137–164.

Lin, X., Tan, K. S., 2003. Valuation of equity-indexed annuities under stochastic interest rates. North American Actuarial Journal 7 (4), 72–91.

Lin, Y., Liu, S., Yu, J., 2013. Pricing mortality securities with correlated mortality indexes. Journal of Risk and Insurance 80 (4), 921–948.

Liu, M., Staum, J., 2010. Stochastic kriging for efficient nested simulation of expected shortfall. Journal of Risk 12 (3), 3.

Ludkovski, M., 2015. Kriging metamodels for bermudan option pricing. arXiv preprint arXiv:1509.02179.

Ludkovski, M., Risk, J., Zail, H., 2016. Gaussian process models for mortality rates and improvement factors.

Marrel, A., Iooss, B., Van Dorpe, F., Volkova, E., 2008. An efficient methodology for modeling complex computer codes with Gaussian processes. Computational Statistics & Data Analysis 52 (10), 4731–4744.

Mitchell, D., Brockett, P., Mendoza-Arriaga, R., Muthuraman, K., 2013. Modeling and forecasting mortality rates. Insurance: Mathematics and Economics 52 (2), 275–285.

Nychka, D., Furrer, R., Sain, S., 2015. fields: Tools for Spatial Data. R package version 8.2-1.
URL `http://CRAN.R-project.org/package=fields`

Oakley, J., 2004. Estimating percentiles of uncertain computer code outputs. Journal of the Royal Statistical Society: Series C (Applied Statistics) 53 (1), 83–93.

Picheny, V., Ginsbourger, D., 2013. A nonstationary space-time Gaussian process model for partially converged simulations. SIAM/ASA Journal on Uncertainty Quantification 1 (1), 57–78.

Picheny, V., Ginsbourger, D., Roustant, O., Haftka, R. T., Kim, N.-H., 2010. Adaptive designs of experiments for accurate approximation of a target region. Journal of Mechanical Design 132 (7), 071008.

Plat, R., 2009. On stochastic mortality modeling. Insurance: Mathematics and Economics 45 (3), 393–404.

Purushotham, M., Valdez, E., Wu, H., 2011. Global mortality improvement experience and projection techniques. Tech. rep., Society of Actuaries.
URL `http://www.soa.org/files/research/projects/research-global-mortality-improve-report.pdf`

Qian, L., Wang, W., Wang, R., Tang, Y., 2010. Valuation of equity-indexed annuity under stochastic mortality and interest rate. Insurance: Mathematics and Economics 47 (2), 123–129.

Rasmussen, C. E., Williams, C. K. I., 2006. Gaussian Processes for Machine Learning. The MIT Press.

Renshaw, A. E., Haberman, S., 2003. Lee–Carter mortality forecasting with age-specific enhancement. Insurance: Mathematics and Economics 33 (2), 255–272.

Renshaw, A. E., Haberman, S., 2006. A cohort-based extension to the Lee–Carter model for mortality reduction factors. Insurance: Mathematics and Economics 38 (3), 556–570.

Revuz, D., Yor, M., 2013. Continuous martingales and Brownian motion. Vol. 293. Springer Science & Business Media.

Riihimäki, J., Vehtari, A., 2010. Gaussian processes with monotonicity information. In: International Conference on Artificial Intelligence and Statistics. pp. 645–652.

Risk, J., Ludkovski, M., 2016. Statistical emulators for pricing and hedging longevity risk products. Insurance: Mathematics and Economics 68, 45–60.

Rosner, B., Raham, C., Orduña, F., Chan, M., Xue, L., Zak, B., Yang, G., 2013. Literature review and assessment of mortality improvement rates in the US population: Past experience and future long-term trends. Tech. rep., Society of Actuaries.
URL http://www.soa.org/Files/Research/Exp-Study/research-2013-lit-review.pdf

Roustant, O., Ginsbourger, D., Deville, Y., 2012a. DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization. Journal of Statistical Software 51 (1), 1–55.
URL http://www.jstatsoft.org/v51/i01/

Roustant, O., Ginsbourger, D., Deville, Y., et al., 2012b. Dicekriging, Diceoptim: Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization. Journal of Statistical Software 51 (1), 1–55.

Salemi, P., Staum, J., Nelson, B. L., 2013. Generalized integrated Brownian fields for simulation metamodeling. In: Proceedings of the 2013 Winter Simulation Conference. IEEE Press, pp. 543–554.

Santner, T. J., Williams, B. J., Notz, W. I., 2003. The Design and Analysis of Computer Experiments. Springer-Verlag, New York, NY.

Santner, T. J., Williams, B. J., Notz, W. I., 2013. The design and analysis of computer experiments. Springer Science & Business Media.

Sfakianakis, M. E., Verginis, D. G., 2008. A new family of nonparametric quantile estimators. Communications in Statistics: Simulation and Computation® 37 (2), 337–345.

Sheather, S. J., Marron, J. S., 1990. Kernel quantile estimators. Journal of the American Statistical Association 85 (410), 410–416.

SOA, 2014a. Mortality improvement scale MP-2014 report. Tech. rep., Retirement Plans Experience Committee, https://www.soa.org/Research/Experience-Study/Pension/research-2014-mp.aspx.

SOA, 2014b. RP-2014 mortality tables. Tech. rep., Society of Actuaries Pension Experience Study, https://www.soa.org/Research/Experience-Study/pension/research-2014-rp.aspx.

SOA, 2015. Mortality improvement scale MP-2015. Tech. rep., Retirement Plans Experience Committee, https://www.soa.org/Research/Experience-Study/Pension/research-2015-mp.aspx.

Sobol, I. M., 1998. On quasi-Monte Carlo integrations. Mathematics and Computers in Simulation 47 (2), 103–112.

Villegas, A. M., Kaishev, V. K., Millossovich, P., 2015a. Stmomo: An R package for stochastic mortality modelling.

Villegas, A. M., Kaishev, V. K., Millossovich, P., 2015b. StMoMo: An R package for stochastic mortality modelling. Tech. rep., SSRN documentation at `papers.ssrn.com/sol3/papers.cfm?abstract-id=2698729`.

Whittaker, E. T., 1922. On a new method of graduation. Proceedings of the Edinburgh Mathematical Society 41, 63–75.

Williams, C. K., Rasmussen, C. E., 2006. Gaussian processes for machine learning. the MIT Press.

Wyss, G. D., Jorgensen, K. H., 1998. A user's guide to LHS: Sandia's Latin hypercube sampling software. Tech. rep., SAND98-0210, Sandia National Laboratories, Albuquerque, NM.

Yang, J., Cox, D. D., Lee, J. S., Ren, P., Choi, T., 2017. Efficient bayesian hierarchical functional data analysis with basis function approximations using gaussian–wishart processes. Biometrics.

Yang, J., Zhu, H., Choi, T., Cox, D. D., et al., 2016. Smoothing and mean–covariance estimation of functional data with a bayesian hierarchical model. Bayesian Analysis 11 (3), 649–670.